

软件测试系列丛书

# 软件测试工程师

新科海学校 刘德宝 编著

中科院新科海学校

二〇〇八年



## 内容简介

本书从软件测试基础理论、测试管理及 BUG 管理工具、Web 测试环境搭建及数据库知识学习、自动化测试工具等几个方面，结合软件行业的实际情况进行了分析和讲解。本书共分为 9 章，整体结构清晰，内容丰富，讲述由浅入深，层层递进。理论与实践相结合。对于希望从事软件测试行业的读者是一本很好的参考书，而且也可以作为对软件测试工作感兴趣的读者的自学用书。

版 权 专 有 侵 权 必 究

---

（内部讲义）

---

出版发行 / 中科院新科海学校

地 址 / 北京市海淀区苏州街 18 号长远天地大厦 A2 座 10 层

邮 编 / 100080

电 话 / 010-82622282, 82622285

网 址 / <http://www.jobedu.com.cn>

经 销 /

印 刷 /

开 本 / 787 毫米×1092 毫米 1/16

印 张 /

字 数 / 千字

版 次 / 2008 年 月第 版 2008 年 月 第 次印刷

定 价 / 50 元



## 序 言

在发达国家，软件测试是一项很受重视的工作。因此，软件测试的专业化水平非常高。很多从事软件测试工作的人，在工作几年后都会成为非常抢手的人才。一些测试人员因经历过一些产品的整个开发过程，并且具备领导测试团队的能力，因而在软件企业中的地位也显得特别突出。

随着国内软件应用与开发的飞速发展，软件用户对软件质量的要求也在不断的提高。如何提高软件运行的可靠性，改进软件过程，进而提高软件产品的质量，已经成为许多研发机构和软件企业迫切关心的问题。软件测试作为保证软件质量的重要手段，越来越受到人们的关注与重视。许多大中型软件企业都建立了专门的软件质量保证体系，组建了自己的测试团队。然而，由于软件测试工作本身的专业特点和软件测试市场需求的迅速升温，导致了软件测试行业专业人员的短缺。给软件企业产品的质量保障工作推进造成了很大的影响。

与同类教材相比，本教材具有一个显著的特点，就是注重经验与实践。编写本教材的时候，咨询了很多公司的测试管理者与项目管理者。因此，在了解一名软件测试人员在实际工作中所必需了解掌握的知识和技能后，编写了本教材。读者在阅读本书后，会对软件测试在整个软件产品开发中的实施过程有很清晰的了解。对软件测试在产品开发过程的各个阶段中的工作内容以及工作重点有一个明确的认识，并了解不同行业软件产品的特点以及测试方法。

在本书编写过程中，孙亚刚、刘伟、张利国、刘向东等老师为本书的大纲制定和内容编写提供了很多宝贵意见和建议，对他们的大力支持表示感谢。

本书中可能存在一些疏漏，希望给予指正，如果我们对本书有任何意见和建议，请 Email 至 [jiaoyan@jobedu.com.cn](mailto:jiaoyan@jobedu.com.cn) 或登录我们的网站 [www.jobedu.com.cn](http://www.jobedu.com.cn)，我们将努力为您提供更完善的服务和支持。

作者

2008 年 8 月



# 目 录

第 1 章 软件测试理论基础 .....	1
1.1 行业背景 .....	1
1.2 软件测试概述 .....	4
1.2.1 软件测试由来 .....	4
1.2.2 软件测试定义 .....	5
1.2.3 软件测试基本分类 .....	7
1.2.4 软件测试人员职业要求 .....	10
1.3 软件测试流程 .....	14
1.3.1 需求测试 .....	14
1.3.2 单元测试 .....	16
1.3.3 集成测试 .....	16
1.3.4 系统测试 .....	17
1.3.5 性能测试 .....	17
1.3.6 用户测试 .....	18
1.3.7 回归测试 .....	18
1.4 软件项目运作流程 .....	19
1.4.1 市场调研 .....	19
1.4.2 可行性研究 .....	20
1.4.3 产品立项 .....	21
1.4.4 需求调研 .....	21
1.4.5 设计开发 .....	22
1.4.6 系统测试 .....	22
1.4.7 产品发布 .....	22
1.4.8 产品维护 .....	22
1.4.9 产品升级 .....	22
1.5 软件测试工作流程 .....	23
1.5.1 测试部门组织结构 .....	23
1.5.2 测试工作流程 .....	26
第 2 章 Windows Server 2003 服务配置 .....	32
2.1 Windows Server 2003 安装 .....	32
2.1.1 虚拟机安装 .....	34
2.1.2 Windows Server 2003 安装 .....	34

---

2.2 Windows Server 2003 部署IIS服务 .....	43
2.2.1 安装IIS服务 .....	44
2.2.2 部署应用程序 .....	48
2.2.3 调试运行IIS服务 .....	57
第 3 章 BUG管理及测试管理工具 .....	62
3.1 Bug管理流程及工具介绍 .....	62
3.2 Bugzilla工具安装配置 .....	64
3.2.1 Bugzilla安装配置 .....	64
3.2.2 Bugzilla使用 .....	88
3.3 测试件管理 .....	88
3.4 TestDirector安装配置及使用 .....	88
3.4.1 TestDirector介绍 .....	89
3.4.2 TestDirector安装 .....	94
3.4.3 TestDirector后台管理 .....	119
3.4.4 TestDirector项目定制 .....	136
3.4.5 TestDirector前台使用 .....	149
3.4.6 TestDirector移植 .....	191
3.4.7 使用技巧 .....	196
3.4.8 TestDirector总结 .....	197
第 4 章 测试文档编写及Web测试环境搭建 .....	199
4.1 测试文档编写 .....	199
4.1.1 测试计划 .....	199
4.1.2 测试方案 .....	200
4.1.3 测试用例 .....	201
4.1.4 测试报告 .....	208
4.2 功能测试方法 .....	210
4.2.1 静态UI测试 .....	211
4.2.2 动态功能测试 .....	213
4.2.3 Bug寻找经验之谈 .....	214
4.3 Web测试环境搭建 .....	220
4.3.1 Tomcat+JDK测试环境搭建 .....	221
4.3.2 Jboss+JDK 测试环境搭建 .....	237
第 5 章 Linux操作系统 .....	243
5.1 Linux起源、特性及应用领域 .....	243
5.2 主流Linux操作系统发行版简介 .....	244
5.3 FC5 安装 .....	245
5.3.1 创建FC5 虚拟机 .....	245



---

5.3.2 FC5 安装 .....	247
5.3.3 FC5 配置 .....	259
5.4 Fedora Core5 基本命令使用 .....	266
5.4.1 SSH Secure Shell Client工具 .....	267
5.4.2 Fedora Core5 命令使用 .....	275
5.5 Linux部署Tomcat+JDK测试环境 .....	297
5.5.1 JDK安装配置 .....	297
5.5.2 TOMCAT安装 .....	301
第 6 章 配置管理及WinCVS工具使用 .....	305
6.1 配置管理理论 .....	305
6.1.1 软件配置管理定义 .....	305
6.1.2 软件配置管理目的 .....	305
6.1.3 软件配置管理发展历史 .....	305
6.1.4 软件配置管理角色职责 .....	306
6.1.5 软件配置管理流程 .....	307
6.2 配置管理工具 .....	310
6.2.1 CVS服务器安装 .....	310
6.2.2 WinCVS工具使用 .....	312
6.4 配置管理应用 .....	323
6.4.1 版本管理 .....	324
6.4.2 变更控制 .....	325
第 7 章 数据库 (SQL Server、Oracle) .....	328
7.1 SQL Server 2000 使用 .....	328
7.1.1 SQL Server介绍 .....	328
7.1.2 服务器管理 .....	330
7.1.3 Transact-SQL语言 .....	337
7.1.4 数据库管理 .....	347
7.2 SQL Server 学习总结 .....	368
7.3 Oracle9i使用 .....	368
7.3.1 Oracle9i简介 .....	368
7.3.2 Oracle安装 .....	369
7.3.3 Oracle服务启动与关闭 .....	379
7.3.4 表空间管理 .....	379
7.3.5 用户管理 .....	388
7.3.6 表创建、修改与删除 .....	392
7.3.7 视图创建、修改与删除 .....	399
7.4 Oracle学习总结 .....	399

---

第 8 章 QTP自动化功能测试工具 .....	401
8.1 自动化功能测试简介 .....	401
8.2 QTP安装 .....	402
8.2.1 QuickTest Professional简述 .....	402
8.2.2 QuickTest Professional界面介绍 .....	405
8.2.3 QuickTest Professional是如何工作的? .....	406
8.2.4 QTP安装 .....	407
8.3 QTP使用 .....	418
8.3.1 QTP教程 .....	419
8.3.2 QTP测试B/S系统实例 .....	419
8.3.3 QTP测试C/S系统实例 .....	429
8.3.4 QTP描述性编程 .....	431
8.3.5 QTP其他知识 .....	434
8.4 QTP学习总结 .....	437
第 9 章 LoadRunner自动化性能测试工具 .....	439
9.1 LoadRunner概要介绍 .....	439
9.2 性能测试理论基础 .....	440
9.2.1 系统用户 .....	440
9.2.2 并发用户数 .....	440
9.2.3 在线用户 .....	441
9.2.4 PV页面浏览量 .....	441
9.2.5 响应时间 .....	441
9.2.6 请求数与点击数 .....	441
9.2.7 吞吐量 .....	442
9.2.8 系统资源耗用 .....	442
9.2.9 性能测试 .....	442
9.2.10 压力测试 .....	442
9.2.11 负载测试 .....	443
9.2.12 容量测试 .....	443
9.3 LoadRunner安装与了解 .....	443
9.3.1 安装LoadRunner .....	443
9.3.2 了解LoadRunner .....	453
9.4 规划测试 .....	456
9.4.1 测试需求分析 .....	456
9.4.2 建模与抽取性能指标 .....	457
9.5 Vuser脚本设计 .....	461
9.5.1 准备工作 .....	461

---

9.5.2 录制脚本 .....	464
9.5.3 分析脚本 .....	474
9.5.4 增强脚本 .....	476
9.5.5 运行脚本 .....	500
9.6 场景设计执行 .....	503
9.6.1 场景设计 .....	503
9.6.2 场景执行 .....	519
9.7 利用Analysis结果分析 .....	536
9.7.1 结果概要 .....	536
9.7.2 并发用户数 .....	540
9.7.3 事务响应时间 .....	540
9.7.4 点击率 .....	542
9.7.5 页面分解 .....	542
9.7.6 系统资源 .....	543
9.8 LoadRunner学习总结 .....	545
附录一 软件测试常用英语词汇 .....	548
附录二 版本说明书模板 .....	552
附录三 性能测试报告实例 .....	553



# 第 1 章 软件测试理论基础

## 本章要点

本章简单介绍软件测试的一些基础知识,如软件测试的基本概念、工作流程等,同时也介绍了一般软件项目的运作流程与测试工作流程,使读者对软件测试有一个初步的了解,并掌握一定的软件测试理论知识,对公司中如何开展测试工作有个总体的印象。

## 1.1 行业背景

近年来,计算机技术不断的发展与广泛的应用,给整个社会带来翻天覆地的变化。各种各样的计算机技术出现我们身边,坐公交刷卡,买衣服上淘宝,买书也可在当当网上买,这些计算机技术给我们带来的便利与我们的衣食住行相关,而对于国家国防来说,卫星导航,火箭发射等等一系列重要的工作,也都离不开计算机的支撑。计算机是由硬件与软件组成的。硬件,就像我们的基础设置,是由专门的厂商去设置制造,而软件也是由专业的人员去开发测试。

时代的发展,使得计算机的应用环境越来越复杂,从而提高了对硬件、软件的质量要求。从软件行业来讲,如何提高软件的质量,一直是当今软件生产活动中的热门话题。软件测试工作对于寻找软件系统中存在的缺陷、保证软件产品的质量,降低企业的生产成本,提高经济效益都具有不可替代的作用。同时,软件测试工作的实施又是一个非常复杂的过程,需要考虑人员、技术、管理、工具等众多因素,这些因素在软件生产活动中起着极其重要的作用,软件测试人员不仅仅要知道“做什么”,还要知道“为什么这么做”,以及“如何做”。随着软件业的发展,对于优秀的测试员的需求也越来越多。国内软件行业的不断发展,国外的外包项目,甚至于本土项目都转移到中国来开发测试。《龙象之争》一书提出了中国软件与印度软件的关系,国外很多的公司考虑将公司的主要业务转移到中国来,中国越来越趋向世界工厂,中国的劳动力成本与他们本国的成本相对来说要少了很多,这样的趋势就带来巨大的就业缺口。最新《中国软件产业发展环境调查报告》预测,未来 5 年,中国的软件市场销售规模将保持 19.4%的复合增长率。不断发展的软件市场,带来了很多的就业机会,在未来的一段相当长的时间内,软件测试工程师作为软件生产活动中必不可少的角色,需求量将会非常大。

目前我国软件测试人才严重匮乏,当前 120 万软件从业者中具备软件测试专业

技术的人员不超过 5 万，人才缺口达到 30 万。“在国内很多软件企业中，软件开发和测试人员数量之比仅能达到 8:1，”清华大学计算机教育专家谭浩强教授介绍说，“而与此同时，国际先进软件企业的同类数据却在 1:1 到 1:2 之间，差距十分明显。”造成这一结果的主要原因是国内软件测试人才教育相对滞后。据统计，目前我国开设计算机类专业的高校已超过 1000 所，而其中设立“软件测试”细分专业的高校几乎没有，绝大多数计算机专业的学生对软件测试都不了解。软件测试的发展陷入人才饥荒的困境。

根据前程无忧（www.51job.com）数据显示，2008 年 3 月全国 IT 行业总的需求量达到了 12 万以上，与 2 月相比增加了近 2 万个职位，占据 IT 行业有效职位数城市榜单前五位的依次是：上海、北京、深圳、广州、杭州等。在企业类型方面，除去其他类型企业，以民营/私营/非上市公司的需求量最大，其次为外商独资（欧美企业）。而纵观 07 年到现今，各大招聘网站如前程无忧、中华英才，每月的职位排行榜中，IT 类的职位几乎月月折桂，软件测试工程师的需求量在 IT 类职位中又占据相当大的比例。种种现象、数据显示，软件测试工程师已经成为选择职业时不可不考虑一门职业了。

在市场需求量不断增大的环境下，软件测试工程师的待遇也在不断提高。据不完全统计，北京地区的薪资趋势大致如图 1-1 所示：

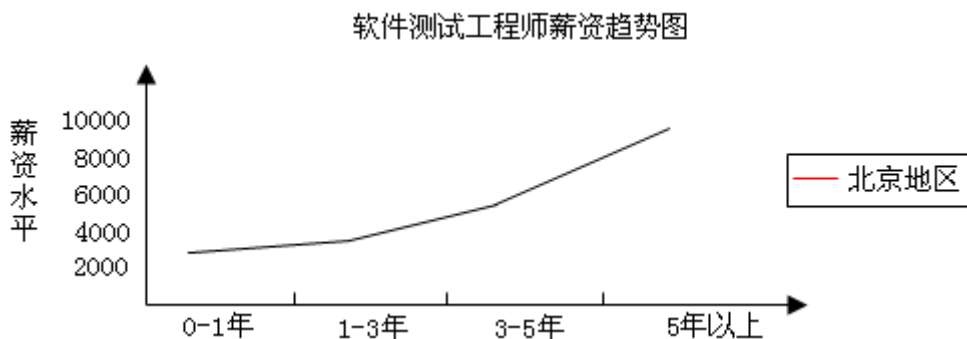


图 1-1 薪资趋势图

测试工程师的薪资高低主要受工作经验、技术能力以及外语因素所制约，而与学历、年龄无关。一般的软件测试工程师，在无多少工作经验，技术能力与外语水平一般的情况下，基本能拿到 2000 到 3000 左右的待遇，随着工作年限的推移以及技术能力的提升，待遇会逐步提升。时间越久，技术能力越深厚的软件测试人员，待遇将与软件开发人员持平，甚至于高过开发人员。所以说，选择软件测试和其他职业一样，都能达到自身的职业要求，甚至于优于其他职业，只要坚持，相信自己的选择，同样可以达到自己心中所期望的高度。

从自身的职业发展来看，软件测试工程师一般有以下几个方向可走：

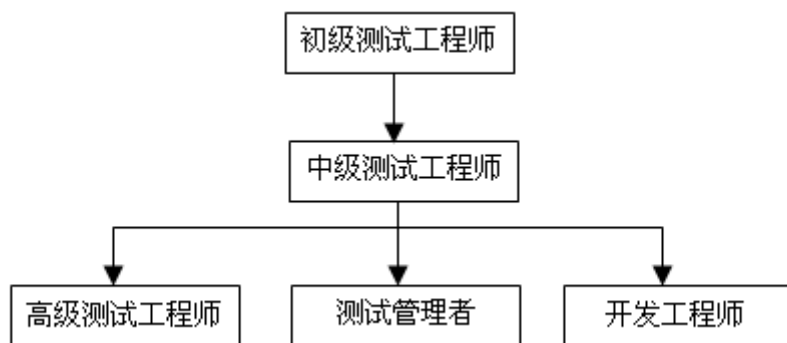


图 1-2 职业发展规划图

如图 1-2 所示，有些软件测试工程师在工作几年后，会从技术路线来规划设计自己的职业发展，也可能选择管理路线，甚至于转向软件开发，这些都是可以的，最重要的是要看自己适应哪种角色，能否坚持。

一个理想的测试人员是应该有开发经验的，至少要有开发的概念。仅仅发现 Bug 是测试的初步，而分析出 Bug 的根本原因，却要有很深的功底。有了开发经历的测试人员工作起来就会更加得心应手，能够容易定位问题、发现潜在的缺陷。软件测试活动中本身就有测试工具的开发以及自动测试的开发，这些脚本的编写比写一个商品化应用系统简单不了多少。而测试人员是开发人员中精通业务的佼佼者，再加上良好的测试观念和开发概念，这些都是一个合格的开发人员应该具备的，从软件测试转为开发，也是一个非常不错的选择。一般企业需要什么样的软件测试人才呢？他们对软件测试人员的技术技能有哪些要求呢？

一般地公司在招聘时都会提出以下几点要求：

- 一年以上测试经验；
- 计算机相关专业大专以上学历；
- 了解软件工程，熟悉软件测试过程和标准、熟悉配置管理技术和工具；
- 能够编制测试计划、设计测试用例、编写 Bug 报告和测试总结报告、使用测试工具、开发测试脚本；
- 熟练使用 Windows 或 Unix 或 Linux 操作系统；
- 熟悉 C、C++、Java、VB、Delphi、C#中的一种以上；
- 熟练使用 SQL Server 或 Oracle 数据库；
- 了解业务领域(ERP、OA、电子商务、税务系统、电信计费系统...)优先；
- 熟练掌握至少一种以上的测试工具，如 TestDirector、QTP、LoadRunner、Robot 等；
- 进取、合作、表达、沟通、责任心、耐心、认真程度。

在这些招聘要求中，我们试着对照一下，自己具备哪些能力呢？

由于新劳动合同法对员工的福利待遇提出了更高的要求，这点使得大多数公司在选择人才的时候更为慎重。企业希望招聘进来的员工能在短时间内适应工作环

境，并投入到工作中去。由于现在职业培训的成本在增加，企业很少愿意对新进的员工进行培训，于是便提高应聘要求，希望应聘者具备一定的能力，以减少公司企业的培训成本。而这些要求对于没有工作经验，或者想转行的人来说，无疑是一个巨大的困难。企业的要求这么高，我们该如何才能克服这种困难，那么就需要我们有勤奋学习的精神，同时也需要一定的学习方法。

本教程将由浅入深，理论和项目案例交互，理论为辅、实践为重的教学方式，浅显通俗的理论解释，结合常见实用的项目实战，使学者具备成为优秀的软件测试工程师的条件，并可与具有丰富实际经验和理论知识的老师进行双向交流，结合软件测试实践中遇到的问题进行研讨。

对于软件测试初学者，我们提供了切合实际、循序渐进的学习线路，读者在实际学习过程中，可参考图 1-3 所示的软件测试学习路线图，从软件测试的理论基础，到项目实战，逐步学会软件测试的知识，掌握一定的技术技能，最终能够胜任公司的软件测试工作。

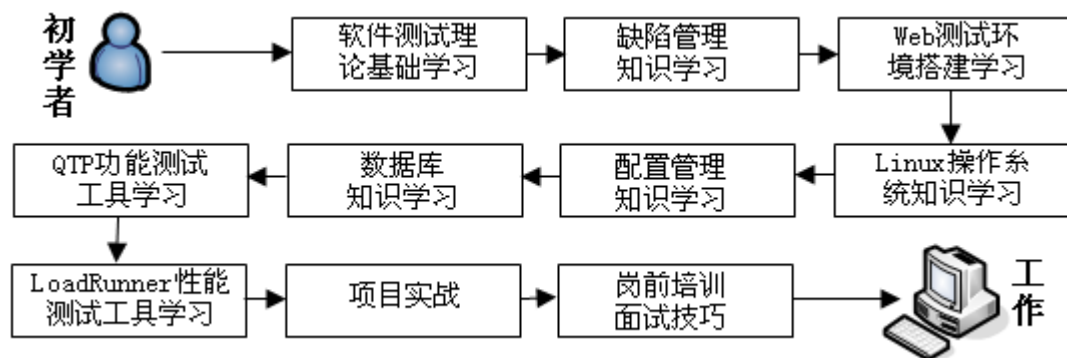


图 1-3 软件测试学习路线图

在学习过程中如果遇到了问题，可以访问 <http://www.v512.com/bbs/index.php> 软件测试频道，将所遇到的问题发布上去，我们会及时给予解答。同时，还提供了教程相关章节的教学视频，可访问 [www.jobedu.com.cn](http://www.jobedu.com.cn) 下载。

## 1.2 软件测试概述

### 1.2.1 软件测试由来

1950 年左右，软件伴随着第一台电子计算机的问世诞生了。上世纪中叶，软件产业从零开始起步，在短短的 50 多年的时间里迅速发展成为推动人类社会发展的龙头产业，造就了一批百万、亿万富翁。随着信息产业的发展，软件对人类社会的发展越来越重要。



过去，软件仅由懂一些开发知识的程序员编写。程序员不仅担负着编写代码的工作，还肩负着程序代码测试、保证代码质量的职责。实际上，程序员此时所做的测试工作并非真正意义上的软件测试。他们所做的测试，从本质上来说，应该称作“调试”。

那么，什么是调试呢？通俗的理解，就是在已知错误的情况下，对软件程序代码做出的一系列检查，校正的过程，而软件测试则是在未知错误的情况下，检查程序代码是否有问题的过程。真正区分测试与调试的意义在于，软件测试从软件质量保证的角度来检查程序代码是否有错误，而调试则是为了解决当前已知的错误，调试活动根本无法替代软件测试活动。以前，在大多数的企业、公司里往往把开发人员的调试过程当作测试，而不招聘专职的软件测试工程师，这样的观点是不正确的。

由于早期的软件只有少量的代码，程序员完全可以应付。开发、调试、直至最后的发布使用。然而，随着真正的商用软件的出现，程序的规模经历了一次又一次的爆炸式的增长。程序规模从最初的几行或几十行类机器语言，到现在的代码行数达到千万数量级，软件的复杂度不断增加，开发的难度也越来越难，随之而来的问题就是如何保证程序的正确性和可用性。此刻，软件再也不是一个只有程序员自己能够理解的黑盒子了，如何在软件程序自身的技术内涵和用户特定领域的需求间找到平衡点，成为学者和实践者们追寻的目标。此时，区别于调试的软件测试活动作为度量软件与用户需求间差距的手段登上了历史舞台。

软件测试活动的出现，解放了程序员，使程序员能够专心的开发代码，优化算法，并能及时的修复测试人员所发现的代码缺陷，提高其工作效率。同时，各司其职的分工方式，也更适合于当今社会的发展模式。

### 1.2.2 软件测试定义

在学习软件测试定义之前，我们先来了解一下什么是软件。

软件这个词在我们的生活中出现的太多太多啦。财务软件，办公软件，聊天软件，播放软件等等，都不陌生，可有多少人能说出什么软件的定义吗？从软件的诞生到现在，经过了半个多世纪，他的发展从早期的简陋模式，到现在复杂模式。说简陋，是因为先前的软件定义仅限于代码，大多数人概念中的软件，是由一些可以运行的，可以实现某些特定功能的代码的集合。这种认识，其实忽略了软件所在的特定的社会环境，在一个真正的商业环境中，我们所生产的不仅仅只是程序代码，而是一个完整的产品，包括实现用户特定需求的程序代码，支撑程序代码运行的一切与之相关的文档，以及程序代码运行所必须的一切数据。就像我们购买一部手机，不仅仅只是手机，还有手机的使用说明书、保修单等等。简单的理解就是软件是一个集合，包含上述的三个部分：程序代码、文档、数据。如图 1-4 所示。

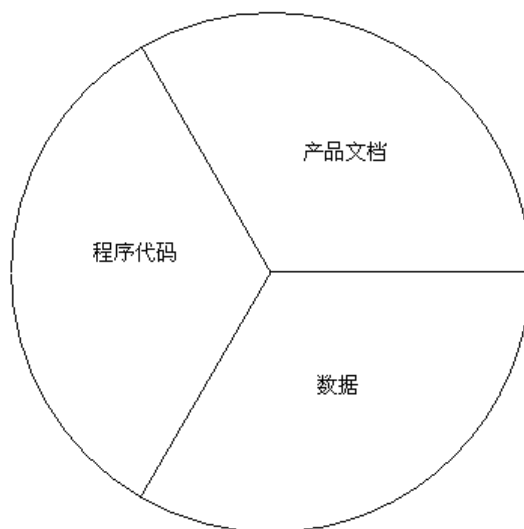


图 1-4 软件构成结构图

在了解软件的定义之后，我们再来学习什么是软件测试，前面介绍了测试与调试的区别，那么到底什么是软件测试呢？

软件测试就是为了发现错误而审查软件文档、检查软件数据和执行程序代码的过程，其目的在于在软件交付使用前充分发现缺陷并协助相关部门定位、解决缺陷，最后交付一个高质量的软件给用户。软件测试是软件生存周期的一个重要阶段，其主要任务是对被测对象，从初期的需求规格说明书，到最后的系统交付使用，整个过程进行检查、验证，以确保软件质量。

基于不同的立场，存在着两种完全不同的测试目的。从用户的角度出发，普遍希望通过软件测试暴露软件中隐藏的错误和缺陷，以考虑是否可接受该产品。而从软件开发者的角度出发，则希望测试成为表明软件产品中不存在错误的过程，验证被测软件已正确地实现了用户的需求，确立人们对软件质量的信心。那么我们软件测试人员在实际的工作过程中应该如何处理呢？

根据软件测试的定义，我们不难得知，软件测试的目的是通过科学的测试方法，找出软件中存在的缺陷，最终得到一个高质量的软件产品，那么确保用户满意将是我们的服务宗旨，如何找到更多的软件缺陷，是我们工作的重点。所以，我们需要从客户的角度出发，按照正确的业务流程尽最大可能去模拟用户的行为习惯，从客户角度出发，找出产品中的缺陷。在此过程中，应以《需求规格说明书》为基本依据，结合软件产品的设计文档，以及项目经验设计高效的测试用例，只有这样，才能达到我们测试的目的。

从广义上讲，软件测试是指软件产品生存周期内所有的检查、评审和确认活动。如：设计评审、文档审查、单元测试、集成测试、系统测试、验收测试等。从狭义上讲，软件测试是对软件产品质量的检验和评价。它一方面检查、揭露软件产品质量中存在的质量问题，同时又需对产品质量进行客观的评价并能提出改进意见。

在软件测试工作中我们经常接触到这样一个名词：Bug。软件测试活动中称寻找软件缺陷的过程为找 Bug，那么到底什么是 Bug 呢？

从字面来看，Bug 就是小虫、臭虫的意思，这跟软件中的缺陷有什么关系呢？这里有个典故。

第一代的计算机是由许多庞大且昂贵的真空管组成，并利用大量的电力来使真空管发

光。可能正是由于计算机运行产生的光和热，引得一只小虫子（Bug）钻进了一支真空管内，导致整个计算机无法工作。研究人员费了半天时间，总算发现原因所在，把这只小虫子从真空管中取出后，计算机又恢复正常。后来，Bug 这个名词就沿用下来，表示电脑系统或程序中隐藏的错误、缺陷或问题。

Bug 的定义很广泛，在软件使用过程中所出现的任何一个可疑问题，或者导致软件不

能符合设计要求或者不能满足消费者需求的问题都是 Bug，即使这个 Bug 在实践中是可行

的。有时候，Bug 并不是程序错误。例如，软件没有按照一般用户的使用习惯来运行，此

时也可以把这个问题看成是该软件的一个 Bug。

从另外一个方面来理解，Bug 就是缺陷、不完美的意思。一切不完美的地方，我们都可以认为其是一个 Bug。

Bug 有很多种分类方法，可以按软件的生命周期分，比如需求分析阶段、产品设计阶段、产品维护阶段等，也可以按照软件的构成分，比如软件文档、软件代码、软件数据等，同样可以按照软件测试的工作点来分，比如 UI 界面、功能、性能等。不管使用哪种分类方法，都是为了有效的区分 Bug 所处的位置，便于尽快定位问题，并解决问题。

### 1.2.3 软件测试基本分类

在了解了软件测试定义后，我们来看看软件测试都有哪些分类。一般地，我们将软件测试活动分为以下几类：黑盒测试、白盒测试、灰盒测试、静态测试、动态测试、手动测试、自动测试等等。

#### 黑盒测试

软件测试行业，最常听到的名词就是黑盒测试，那么到底什么是黑盒测试呢？

黑盒测试又叫功能测试、数据驱动测试或基于需求规格说明书的功能测试。该测试类别注重于测试软件的功能性需求。

采用这种测试方法，测试工程师把测试对象看作一个黑盒子，完全不考虑程序内部的逻辑结构和内部特性，只依据程序的需求规格说明书，检查程序的功能是否符合它的功能说明。如图 1-5 所示，测试工程师无须了解程序代码的内部构造，

完全模拟软件产品的最终用户使用该软件，检查软件产品是否达到了用户的需求。举个例子，我们购买了手机以后，很少有人打开手机观察其内部的结构，大多数情况下，我们只是使用该手机的功能，从某种意义上说，此时这部手机就是我们的测试对象，所采用的测试方法就是黑盒测试。

黑盒测试方法能更好更真实的从用户角度来考察被测系统的功能性需求实现情况。在软件测试的各个阶段，如单元测试、集成测试、系统测试及确认测试等阶段中都发挥着重要作用，尤其在系统测试和确认测试中，其作用是其他测试方法无法取代的。



图 1-5 黑盒测试示例图

### 白盒测试

与黑盒测试相对的软件测试方法，称为白盒测试。白盒测试又称结构测试、逻辑驱动测试或基于程序代码内部构成的测试。此时，测试工程师将深入考查程序代码的内部结构，逻辑设计等等。就像前面的例子，我们打开手机，观察手机电路板的设计，液晶屏的构成等等。白盒测试需要测试工程师具备很深的软件开发功底，精通相应的开发语言，一般的软件测试工程难以胜任该工作。图 1-6 是白盒测试的示例图，相对于白盒测试工程师来说，软件产品的内部构成是敞开的。

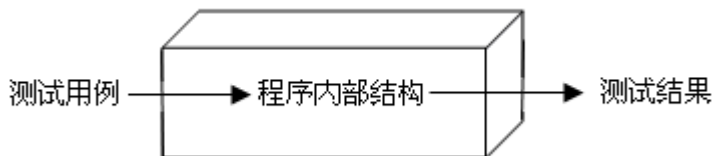


图 1-6 白盒测试示例图

### 灰盒测试

与前面的黑盒测试、白盒测试相比，灰盒测试介于两者之间。黑盒测试仅关注程序代码的功能性表现，不关注内部的逻辑设计、构成情况，白盒测试则仅从程序代码的内部构成考虑，检查其内部代码设计结构，方法调用等。而灰盒测试结合这两种测试方法，一方面考虑程序代码的功能性表现，另一方面，又需要考虑程序代码的内部结构。通俗的讲，灰盒测试就是白加黑，像我们的性能测试，自动化功能测试就采用了灰盒测试的方法。图 1-7 是灰盒测试的示例图。



图 1-7 灰盒测试示例图

### 静态测试

静态测试，顾名思义，就是静态的、不执行被测对象程序代码而寻找缺陷的过程。通俗的讲，静态测试就是用眼睛看，阅读程序代码，文档资料等，与需求规格说明书中的客户需求进行比较，找出程序代码中设计不合理，以及文档资料有错误的地方。

一般在企业、公司里会召开正规的评审会，通过评审的方式，找出文档资料、程序代码中存在缺陷的地方，并加以修改。

在进行代码的静态测试时，可以采用一些代码走查的工具，如 QA C++、C++Test 等。

### 动态测试

动态测试即为实际的执行被测对象的程序代码，输入事先设计好的测试用例，检查程序代码运行得到的结果与测试用例中设计的预期结果之间是否差异，判定实际结果与预期结果是否一致，从而检验程序的正确性、可靠性和有效性，并分析系统运行效率和健壮性等性能状况。

动态测试有四部分组成：设计测试用例、执行测试用例、分析比较输出结果、输出测试报告。

动态测试有三种主要的方法：黑盒测试、白盒测试以及灰盒测试。

### 手动测试

在未真正接触软件测试之前，很多人都认为，软件测试工作就是执行一些鼠标点击的动作来查找缺陷。的确，在手动测试阶段，大部分的测试工作就是模拟用户的业务流程，来使用软件产品，从而发现软件产品中的缺陷。手动测试是最传统的测试方法，也是现在大多数公司都使用测试形式。它是测试人员设计测试用例并执行测试用例，然后根据实际的结果去和预期的结果相比较并记录测试结果，最终输出测试报告的测试活动。这样的测试方法，可以充分发挥测试工程师的主观能动性，将其智力活动体现于测试工作中，能发现很多的缺陷，但同时这样的测试方法又有一定的局限性与单调枯燥性。

### 自动化测试

随着软件行业的不断发展，软件测试技术也在不断地更新，出现了众多的自动

化测试工具，如 HP 的 QucikTestProfessional、LoadRunner，微软的 WAS，IBM 的 Rational 一套等等。所谓的自动化测试，就是利用一些测试工具，模拟用户的业务使用流程，让他们自动运行来查找缺陷。也可以编写一些代码，设定特定的测试场景，来自动寻找缺陷。自动化测试的引入，大大的提高了测试的效率和测试的准确性，而且写出的比较好的测试脚本，还可以在软件生命周期的各个阶段重复使用。

自动化测试的优点是能够很快、很广泛地查找缺陷，同时可以做很多重复性的工作，在回归测试阶段，我们可以利用 QuickTestProfessional 自动化功能测试工具进行，而无须大量的软件测试人员手动的再次执行测试用例，极大的提高了工作效率。有时候我们常常需要做一种压力测试，需要几万甚至几十万个用户同时访问某个站点，以保证网站的服务器不会出现死机或崩溃的现象。一般来说，要几万人同时打开一个网不现实的，就算能够找到那么多的测试者，成本也很高。但是，利用测试工具，比如 LoadRunner，就可以非常容易地做到，并且测试工具还可以自动判断浏览结果是否正确。

然而，自动化测试的缺点也很明显，它们只能检查一些比较主要的问题，如崩溃、死机，但是却无法发现一些一般的日常错误，这些错误通过人眼很容易找到，但机器却往往找不到。另外，在自动测试中编写测试脚本工作量也很大，有时候该工作量甚至超过了手动测试的时间。

如今很多的软件产品生产，我们都是追求短（项目周期短）、平（成本低）、快（市场反应快），根本没有那么多的时间留给我们做测试，在这种情况下，如果我们不识适宜的盲目追求自动化测试，可能带来的不是好处，而是项目的失败。

需要注意的是，在自动化测试活动中，测试工具的应用可以提高测试的质量、测试的效率。但是在选择和使用测试工具的时候，我们也应该看到，在测试过程中，并不是所有的测试工具都适合我们使用，同时，有了测试工具、会使用测试工具并不等于测试工具真正能在测试中发挥作用。因此，我们应该根据实际情况选择或者不选择测试工具，选择使用何种测试工具，千万不能为了使用工具而刻意的去使用工具。

#### 1.2.4 软件测试人员职业要求

人力资源是各项工作中最重要也是最宝贵的资源，测试工作也同样。一个项目想开展有效的测试活动，就需要有一个积极的、合格的测试组。

在国内软件测试刚起步的时候，有一个普遍的现象就是让那些经验最少的新手去做测试工作。没有人愿意做测试，公司觉得养测试人员是一件高成本的事情，不合适。但实际上测试工作是一项重要的工作，因为测试部门是软件发布的出口，又可能是客户意见反馈的入口。如今，这种观点已经改变，很多公司都需要优秀的软件测试工程师，然而，因为以前的不重视，导致了优秀的软件测试工程师非常难得。经验表明，对系统进行有效的测试所需要的技能绝对不比进行软件开发需要的少。测试工程师需要对系统有较全面的了解，而开发者可能只对自己的模块理解的较深

入。一个优秀的测试工程师要对于一些不易重复出现的错误找到规律，要能够帮助开发人员定位问题，能够对代码进行一定的检查，将错误尽可能的在项目测试的早期阶段发现，同时，测试工程师还要对各种编程语言、数据库都有一定的了解，有编程的概念。下面，我们从个人素质要求角度来阐述一下软件测试工程师所需具备的素质。

## 1. 责任心

由于目前的软件测试行业处于初级阶段，还没有很好的量化指标对软件测试活动做出衡量。有些企业、公司将测试工程师发现缺陷的多少作为绩效考核的指标，然而这种方法有很大的弊端，软件测试工作本身就是一个主观色彩很强的工作，测试工程师在测试活动中需尽可能地模拟软件产品最终用户的业务流程来进行测试，但实际工作中，是不太可能做到的。大多数情况下，测试结果都是基于测试工程师根据项目文档和自己对软件产品的理解基础上得出的。

我们知道，在我们思维定势的时候，即使是再简单的错误，我们也可能无法找到。俗语说“当局者迷，旁观者清”，我想用在这里一点都不为过。在测试工作开展的初期，可能软件产品中存在大量的缺陷，即使测试工程师不花费太多的精力，就可以找到很多的缺陷，而随着测试工作的不断深入，即使测试工程师再怎么用心，也不一定能找到多少缺陷。所以，以缺陷的多少来衡量软件测试工程师的工作质量，并非一种明智、公平的考核方法。

我在测试工作中就遇到这样的问题，记得当时的总监跟我说，对软件测试人员的要求是每天至少提交 8 个 bug，我认为这样的量化指标是不合理的。在实际工作中，软件测试活动在很大程度上依赖于测试人员的责任心，能够主动去做测试去完成工作，而无法绝对的以缺陷的多少来衡量。所以，就需要我们软件测试人员有极强的责任心，确保自己的工作质量。

## 2. 沟通能力

说到沟通，可能大家都会说“这个简单，我沟通能力不错，八面玲珑，什么人我都能搞定”。其实，在我们软件测试活动中，沟通能力并不是我们通常意义上所讲的交流，这里包含的更多成份是技术含量以及服务意识。

这里的沟通不是耍嘴皮，测试是连接开发和客户的接口，与开发人员沟通，我们需要从专业知识角度考虑，比如当我们发现的缺陷开发人员不认可的时候，我们如何从理论、实际应用以及缺陷可能引发的后果等角度去阐述缺陷，使他们认同我们的观点。所做出的阐述要有理有据，而不是强词夺理，更不是争吵。

实际工作中，开发人员与测试人员在某种角度上来讲是对立的。表面看来，软件测试的目的和软件工程活动的所有其他的工作目的都是相反的：其他工作是“建设性”的，而测试工作是“破坏性”的——尽最大可能证明程序中有错误，不能按照预定的要求正确工作。从这点来说，软件测试与软件开发是相对的，有冲突的。

就像生活中，如果有一帮人专门挑你的刺，不知道你的感受如何。所以，在这个角度上看，开发人员与测试人员是对立。不过，这仅仅是从表面来看。实际上，发现问题，揭露问题并不是软件测试的最终目的。发现问题是为了解决问题，软件测试的根本目标是尽可能多、尽可能早地发现软件生产过程中的问题，并与其他部门一起定位问题、排除问题，最终把一个高质量的软件系统交给用户使用。从这点来说，软件测试与软件开发又是统一的，所以软件测试与软件开发从整个软件生产过程来看他们是一个利益的共同体，只是在这个过程中扮演了两个不同的角色。

一句话，我们的共同目标都是为了提交一个高质量的软件产品给用户，所以我们在实际工作中需要将我们最大的可能去理解对方，提高双方的工作效率。

另外，测试人员有时又需要跟我们的客户进行交流，从非技术的角度出发，能将枯燥的、难懂的技术概念，介绍给客户，帮助他们理解软件、掌握软件的使用，并从系统的实际使用角度来考虑问题，帮助客户解决，又不触犯企业、公司的利益。好的测试工程师能在这两种环境中应付自如，不仅提高了自身的工作效率，也为企业、公司赢得了客户满意度。

### 3. 团队合作精神

在日常生活中，我很喜欢篮球运动，喜欢这种体育运动中所体现出来的团队合作精神。团队配合赢球后的感觉让我很开心、很有成就感。比赛、娱乐过程中，每时每刻，都不是一个人的战斗。一个球队不是一个人的球队，而一个部门也不是一个人的部门。我们知道，在实际的软件产品生产过程中，需要很多部门协同工作，市场部、研发部、测试部、售后服务部、运维部等等。一个高质量的软件产品从设计到生产到发布，是众人努力劳动、智慧的结晶。单枪匹马闯天下的局面已经不复存在。在我的工作经历中，每家公司都在强调这种精神。比如华为公司，在新员工入职培训中，会通过多种形式，如演讲、相声、小品等，来培养、增强我们的团队合作意识。软件测试工作从其工作内容来看，是极具破坏性的工作，而开发则是建设性的，从这点可以看出，如何做沟通，如何很好的协调我们的工作关系，将决定着我们工作的质量。所以，软件测试人员需要具备高度的团队合作精神，与其他同事，一起努力，为保证软件产品的质量做出努力。

### 4. 耐心、细心、信心

女生与生俱来的细心，将在软件测试工作中体现无遗，而男生的信心也将在此刻得到充分发挥。软件测试工作中，我们需要有极大的耐心、细心、信心。很多人在不了解软件测试真正的工作内容时，会认为这项工作是枯燥的、无味的。其实不然，每个人工作需要一种成就感，这种成就感让我们时刻保持着工作的激情。当我们设计了比较高效的用例，在软件产品测试初期，发现了大量缺陷的时候，大家心里都被幸福装的满满的。然而，这只是开始，随着测试的深入，发现缺陷越来越难，不是所有的缺陷都能容易的找出，这个时候，就需要我们耐心、细心了。黄豆里面



挑芝麻，需要我们花费更多的时间，更多的精力去发现、分辨、识别和分派一个缺陷。

人无完人，在项目管理不规范，文档不齐全等等客观因素的影响下，增加了软件测试工程师犯错的机率。因为对需求的误解，业务知识的缺乏等等，可能会提出一些不是缺陷的缺陷，那么此时，一直被我们“挑刺”的开发人员，就会“适时”给予“打击”。这个时候，测试人员应对自己的观点有足够的自信心，并能谦虚的接受别人对自己提出的意见和建议，提高自身发现问题并解决问题的能力，以及沟通表达的能力。而不是只感到委屈，影响心情，有时候女孩子可能在这方面表现较为突出一些，心情不好，甚至“伤心落泪”，其实这些都是没有必要的。对自己要有信心，不足的地方就去学习、弥补，而不是停滞不前、妄自菲薄。

### 5. 时时保持怀疑态度，并且有缺陷预防的意识

软件测试与软件开发的工作内容不一样，这就导致了在某些时候，当测试工程师发现了缺陷后，开发人员会以各种借口将该缺陷掩饰过去，测试工程师则需要从理论技术、实际用户需求角度出发，采用一定的策略去准备各种测试数据，从每个细节上设计不同的应用，去证明缺陷确实存在，或者确实可能造成比较大的不良后果。不去想当然的假定任何一个数据是可行的，从事实出发，尽可能多的找出软件产品中的缺陷，并协助开发人员定位问题，以求解决问题，这样才能不断的发现问题、解决问题。测试工程师要有怀疑一切的态度，不为每一次的“狡辩”而放弃自己的立场。记住，一切以数据说话！

软件测试工程师作为软件质量保证的一部分，一定要有质量防范的意识，也就是要有缺陷预防的意识。当曾经发生过的缺陷再次发生后，我们需要研究它出现的原因，找出解决办法，从而避免再次出现。要善于分析测试结果，缺陷分布情况等。不断的总结，加强缺陷预防的能力，提高软件测试的工作效率。

### 6. 不断学习的能力

大多数企业、公司在招聘软件测试工程师时，提出了很高的要求，开发语言掌握情况，测试理论熟悉程度，测试工具使用经验等等。为了获得更多的发展机会，提高自身的职业竞争力，软件测试工程师需要不断地学习，掌握开发人员所使用的编程语言，能更多、更有效找出缺陷，掌握一定的测试理论、测试工具将会大幅度提高自身的工作效率。同时，也为自身的发展奠定坚实的基础。

软件测试工作其实对软件测试工程师的技能要求很高，比如说编程语言，不一定要精通，但我们必须要了解大部分的编程语言，要具备软件开发的思想。同时，测试工程师还要掌握众多的业务知识。所以，学习能力对测试工程师来说是非常重要的。

## 1.3 软件测试流程

软件测试虽然是软件生存周期的一个独立阶段，但测试工作却渗透到从分析、设计直到编程的各个阶段中，如测试计划的编写从分析和设计阶段就开始了，而具体的测试工作随编程工作的不断深入也在进行中。在实际工作中，测试环节可分为明显的、同等重要的几个阶段：即需求测试、单元测试、集成测试（又称组装测试）和系统测试。测试工作中的第六个阶段是验收测试阶段，验收测试无论在规模上或性质上都和系统测试很相似，它们的根本区别在于：前者是内部的，而后者则是受“客户”控制的。图 1-8 是软件测试所经阶段的一般流程。



图 1-8 软件测试流程图

### 1.3.1 需求测试

软件测试 V 模型要求我们在需求阶段就开始制定系统测试的计划，开始考虑系统测试的方法，但这还是不够的。全面的质量管理要求我们在每个阶段都要进行验证和确认的过程。因此在需求阶段我们还需要对需求本身进行测试。这个测试是必要的，因为在许多失败的项目中，70%~85%的返工是由于需求方面的错误所导致的，并且因为需求的缘故而导致大量的返工，造成进度延迟、缺陷的发散，甚至项目的失败，这是一件极其痛苦的事情。因此我们要求在项目的源头（需求）就开始测试。

在接到测试项目的前期，我们需要对被测软件的需求规格说明书、概要设计文

档、详细设计文档、数据库设计文档等文档资料进行查阅，重点检查需求规格说明书中是否存在描述不准确、需求定义模糊、需求用例不正确、语言存在二义性等等问题。如果不对这些文档进行检查，将可能在后期的测试工作中不断接到系统修改、需求变更等问题。

需求测试我们主要从以下几个方面考虑：

### **完整性**

每一项需求都必须将所要实现的功能描述清楚，从而为开发人员设计和实现这些功能提供所有必要的需求依据。

### **正确性**

每一项需求都必须准确地陈述其要开发的功能。

### **一致性**

一致性是指与其它软件需求或高层（系统，业务）需求不相矛盾，或者与我们的项目宣传资料要一致。

### **可行性**

每一项需求都必须是在已知系统和环境的权能和限制范围内可以实施的。

### **无二义性**

对所有需求说明的读者都只能有一个明确统一的解释，由于自然语言极易导致二义性，所以尽量把每项需求用简洁明了的用户性的语言表达出来。

### **健壮性**

需求的说明中是否对可能出现的异常进行了分析，并且对这些异常进行了容错处理。

### **必要性**

“必要性”可以理解为每项需求都是用来授权你编写文档的“根源”。要使每项需求都能回溯至某项客户的输入，如需求用例或别的来源。

### **可测试性**

每项需求都能通过设计测试用例或其它的验证方法来进行测试。

### **可修改性**

每项需求只应在 S R S（软件需求规格说明书）中出现一次。这样更改时易于保持一致性。另外，使用目录表、索引和相互参照列表方法将使软件需求规格说明书更容易修改。

### 1.3.2 单元测试

单元测试又称为模块测试，顾名思义，就是对程序代码中最小的设计模块单元进行测试。单元测试是在软件开发过程中要进行的最低级别的测试活动。在单元测试活动中，我们主要采用静态测试与动态测试相结合的办法。首先采用静态的代码走查，检查程序代码中不符合编程规范，存在错误或者遗漏的地方，同时使用代码审查的方法，小组检查项目代码，以期发现更多的问题，然后再使用单元测试工具，比如 JUnit 等工具进行程序代码内逻辑结构、函数调用等方面进行测试。据业界统计，单元测试一般可以发现大约 80% 的软件缺陷。

单元测试对测试人员的要求相对较高，一般需要有几年的代码编写经验，并且要十分熟悉当前的被测系统，以及该系统是否与其他系统的接口关联情况。在大多数公司中，单元测试一般情况下有对应的开发工程师负责。

在实际的工作过程中，很多人认为单元测试没有必要，认为这个过程耽误了工作进度，没什么实际的效果，其实不然，软件测试活动贯穿于软件生产的整个过程，每个环节的检查测试都是不可缺少的，我们知道缺陷发现的越早，其修复的代价也就越小。单元测试在编码阶段占据着非常重要的地位。编码只是一方面，还需检查编码，保证代码的质量，所以说，在软件生产过程中及时的开展单元测试是非常有必要的，可以降低编码的错误率，提高编码质量。

### 1.3.3 集成测试

我们知道，个体不代表全部，在单元测试阶段，发现并解决了部分的问题，但不能解决所有的问题。在当前环境下，我们可能觉得代码质量已经非常好了，至少我们的能力范围内不能再发现缺陷了。此时，仅仅是一个代码模块，功能模块，一旦组合起来，可能相互之前的问题暴露了。

集成测试，又称为组装测试，就是将软件产品中各个模块组装起来，检查其接口是否存在问题，以及组装后的整体功能、性能表现。在开展集成测试之前，我们进行了深入的单元测试（当然，实际工作中大多公司不会做单元测试，仅有程序员各自检查自己的代码），从个体来讲，可能解决了很多的缺陷，但所有的个体组合起来，就可能出现各种各样的问题。 $1+1<2$  的问题，此刻尤为突出。

在单元测试阶段，我们无法发现资源争用、接口调用、时钟延迟等等问题。假如有两个模块 A、B，一个数据在 A 处处理时因代码问题，延迟了 0.01 秒，然后流转到 B 模块处理，又延迟了 0.01 秒，那么对于个体来讲，可能 0.01 秒算不了什么，但当数据流转的环节增加时，相应的延迟时间也在不断的增加，最终的累加数据可能带来非常严重的后果，二战期间的爱国者导弹时间，就给了我们深刻的教训。每个环节的缺陷在最终被放大后，可能会引起软件的失效。所以，单元测试阶段的成果并不能保证集成测试没有问题。采用科学有效的集成测试方法，在软件生产活动中是非常有必要的。

集成测试一般可采用非增式集成方法、增式集成方法（自底向上集成；自顶向下集成；组合方式集成）等策略进行测试，利用以黑盒测试为主，白盒测试为辅的测试方法进行测试。集成测试人员一般有测试工程师担当，开发人员将经过单元测试的代码集成后合成一个新的软件版本，交由配置管理员，然后测试组长会从配置管理员处提取集成好的版本进行测试。就像一个自行车生产线一样，工人将自行车的各个部件先检查，没有问题后再组装成一辆完整的自行车交由质检人员进行检查。软件测试工程师就像这里的质检人员，负责对集成组装后的软件版本进行检测。

集成测试阶段主要解决的是各个软件组成单元代码是否符合开发规范、接口是否存在问题、整体功能有无错误、界面是否符合设计规范、性能是否满足用户需求等等。

### 1.3.4 系统测试

系统测试，是将通过集成测试的软件，部署到某种较为复杂的计算机用户环境进行测试，这里所说的复杂的计算机用户环境，其实就是我们一般用户的计算机环境。比如，我们开发了一套财务软件，软件中采用了 EXCEL 2000 版中的某些绘图控件，在集成测试阶段，我们大多数是在一种比较干净的系统进行测试。所谓的干净，就是在测试机上没有多余的软件，仅有所需的操作系统和被测软件。当我们集成测试完成后，就将被测软件置入比较复杂的运行环境下，进行集成和确认测试。在这个过程中，我们往往有很大的收获，比如进行安装测试的时候，会发现在集成阶段安装没有问题，而在复杂的用户环境下，却不能安装。举个例子，前面所说的财务软件，当我们的计算机中装有新版本的 EXCEL 时，如 EXCEL 2003，就可能出现无法正常使用该软件的问题，因为该财务软件所需的控件在 EXCEL 2003 可能不存在，或者不适用。类似于这样的问题，就需要我们将被测软件置入一个较为普遍的用户计算机环境中。

系统测试的目的在于通过与系统的需求定义作比较，发现软件与系统的定义不符合或与之矛盾的地方。这个阶段主要进行的是安装卸载测试、兼容性测试、功能确认测试、安全性测试等等。系统测试阶段采用黑盒测试方法，主要考查被测软件的功能与性能表现。如果软件可以按照用户合理地期望的方式来工作的时候，即可认为通过系统测试。

系统测试过程其实也是一种配置检查过程，检查在软件生产过程中是否有遗漏的地方，在此时做到查漏补缺，以确保交付的产品符合用户质量要求。

### 1.3.5 性能测试

软件测试工作对于一般的软件产品而言，主要测试四个方面：文档、界面、功能、性能。前面我们大部分篇幅描述的多是文档、界面与功能方面的测试。这里，我们来了解一下性能测试的概念。

性能测试所包含的内容比较多，按照典型的概念来理解，就是要求被测软件在

业务处理速度,处理能力和所耗用的硬件系统资源比率满足用户的需求。举个例子,对于某个论坛,我们需要测试该论坛支持 10000 个用户的同时使用,并且在这种情况下,打开帖子的速度能否控制在 4 秒钟以下,论坛服务器的 CPU 使用率不超过 80%,内存占用率不超过 75%等等,这些就是典型的性能测试指标。实施性能测试,一方面可以验证被测软件是否符合用户需求,另一方面,可以得到相关的性能数据,为被测软件的优化提供参考。

随着 B/S 结构软件流行,越来越多的用户希望他们的软件能够有比较好的性能表现,于是性能测试工作的地位被提升,重要性也日渐显露。那么具体如何实施性能测试呢?

我们知道,计算机的处理速度是非常快的,如果想考查某项功能是否达到我们的需求,比如业务处理速度,我们不可能手动记录处理时间,因为这样是不现实的,也是不明智。同时,有些时候我们需要模拟大量用户使用系统的情况,比如测试新浪网能否承受 100 万网络用户的并发访问,那么是不是要找 100 万个人,100 万台电脑在同一时刻登录新浪网呢?显然不现实,也不太可能。一系列实际的问题告诉我们,不要尝试用手动方式进行性能测试,应当编写一段相应的程序或者使用专门的工具进行,比如利用 LoadRunner 自动化性能测试工具。

性能测试的难度相对来说比较大,要求测试人员掌握编程语言,精通业务流程,拥有深厚的项目经验。所以,想顺利的开展性能测试,需要测试工程师不断学习,掌握相应的知识。

### 1.3.6 用户测试

在系统测试完成后,将会进行用户测试。这里用户测试,其实可以称为用户确认测试。在正式验收前,需要用户对本系统做出一个评价,用户可对交付的系统做测试,并将测试结果反馈回来,进行修改、分析。面向应用的项目,在交付用户正是使用之前要经过一定时间的用户测试。

用户测试在整个软件生产流程中非常重要,这个环节是被测软件首次作为正式的系统交由用户使用,用户会根据他们的实际使用情况进行测试,试用,并提出实际使用过程中的问题。我们知道,软件测试是尽可能的去模拟客户的业务行为,遵循既定的用户需求和软件生产规范,寻找软件产品中的缺陷。然而,测试工程师并不是真正的最终用户,所以,在测试过程中仍旧会存在一些未能发现的实际业务缺陷,这对软件质量的保证并不是一个好消息。所以,在产品正式发布前,加入用户的测试是一个明智的选择,因为用户能从最终的业务角度来试用系统,并能发现很多有价值的缺陷,从某个角度来说,用户测试是软件生产流程中的最后质检关。

### 1.3.7 回归测试

简单说,回归测试就是过一段时间以后再回过头来对以前修复过的 Bug 重新进行测试,看该 Bug 是否会重新出现。

回归测试一般发生的情况在用户发现缺陷反馈到公司后,测试部门将组织一次回归测试。或者,公司内部人员发现某软件存在某种缺陷的时候,也可以发起回归测试。回归测试阶段主要的目的是检查以前的测试用例能否再次通过,是否还有需要补充的用例等等。

有些公司会采用自动化测试工具来进行回归测试,比如利用 QTP,对于产品级,变动量小的软件而言,我们可以利用这样的工具去执行测试。但一般情况下,都由测试工程师手动的执行以前的测试用例,来检查用例通过情况。

回归测试可以发现在产品发布前未能发现的问题,比如时钟的延迟问题,软件的性能问题等等。

## 1.4 软件项目运作流程

一般的,在企业或公司的内部,软件项目的运作流程如图 1-9 所示。该图展示了一个软件产品的生命周期。其实与大多数的事情一样,软件也是有生命。从初期的市场调研,直至最后的产品升级换代,就像我们人类的生命周期一样,从出生到哺育后代,经历一个漫长的过程。那么在这些不同的阶段,我们都做些什么事情呢?软件产品又是怎样生产出来的呢?知其然,且知其所以然,作为软件测试工程师,如果不知道一个软件产品是如何生产出来,就显得不专业了。

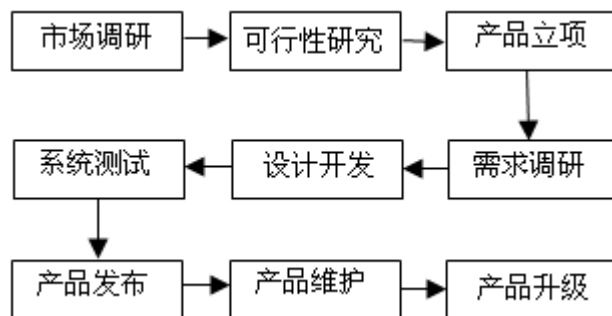


图 1-9 软件项目运作流程图

软件产品的生产经过了各个阶段,接下来将简要的阐明一下各个阶段的工作事项,以期说明软件产品是如何生产出来的。

### 1.4.1 市场调研

市场调研主要有两种形式:一种是主动模式,另外一种是被动模式。首先我们来看看主动模式。

## 主动模式

所谓主动模式，是将公司或者企业作为需求接收的被动方，而需求的提出者作为主动方。这种模式中，需求提出者有明确的主体，如果 A 公司需要 B 公司为其开发一套企业信息管理系统，那么 A 公司为需求提出者，主动方，B 公司为需求接收者，被动方。

这种模式下进行需求调研，一般来说比较容易处理，同时因为有明确的对象，总体项目收益较被动模式明显。比如，华为公司为中国移动广东公司所做业务运营系统，中国移动广东公司是需求的提出者，主动方，而华为公司则是需求接收者，被动方。当项目开发过程中，有不明确、疑问的地方，华为公司可直接咨询移动公司，以便得到更详细，更准确的需求。这种模式属于一对一的模式。

## 被动模式

所谓被动模式就是在没有明确的需求提出者时，由公司或企业主动提出给特定使用用户群提供某种产品的模式。如某公司发现在邮箱业务上存在大量的潜在客户，那么可由公司的市场人员，主动提出此类需求。

这种模式较主动模式有明显的差异，在用户群不确定的情况，需求将是不明确的，给公司或企业的带来一定风险，后期的项目运作过程可能面临着多次变更的风险。比如，今年的奥运会，有些门户网站发现如果在奥运会期间，提供网络直播，网友互动的平台，可能为公司带来名誉或者经济上的利益，那么则会开发相应的平台。这时，所面向的用户群体就是广大的网络用户，而无特定明确的主体，这种项目风险较大，甚至于失败。2004 年的世界杯，搜狐就在视频播放这一领域大赚一笔，打败了新浪。这是成功的例子，也有失败的，前些年比较流行的网络交友，如世纪佳缘，做的非常成功，于是也有很多公司、网站开展这样的业务，但最后因市场的饱和及管理问题，绝大多数都失败了。所以这种模式下，风险非常大。被动模式属于一对多的模式，这种模式需要市场人员，管理人员有很好的市场洞察力及发展眼光，才能在一定程度上保证项目成功。

- 市场调研主体：市场人员、销售人员等
- 调研方式：客户走访，市场观察，报刊媒体等
- 输出文件：《XXX 项目市场调研分析报告》

### 1.4.2 可行性研究

可行性研究方法是以前预测为前提，以投资效果为目的，从技术上、经济上、管理上进行全面综合分析研究的方法。可行性研究的基本任务，是对新开发产品或升级产品的主要问题，从技术经济角度进行全面的分析研究，并对其投产后的经济效益进行预测，在既定的范围内进行方案论证的选择，以便最合理地利用资源，达到预定的社会效益和经济效益。

从软件角度看，可行性研究重点解决的是前期的市场调研是否可行，能否在



一定的成本压力下，持续的为公司、企业带来适当的利益，无论从社会效益还是经济效益。

- 可行性研究主体：市场人员、销售人员等
- 可行性研究对象：在市场调研阶段产生的《XXX 项目市场调研分析报告》
- 输出文件：《XXX 项目可行性分析报告》

### 1.4.3 产品立项

在前期的市场调研、可行性研究经过评审确认可行后，则由需求调研人员牵头，进行产品立项，并进行产品小组的建立，同时制定产品的运作计划，如需求调研，产品设计，产品测试，产品发布等等一系列的工作步骤及时间点。

- 立项负责人：市场调研人员
- 工作内容：提交产品立项申请，审批通过后，制定产品计划书，确定产品各个阶段的工作流程及时间进度表。

### 1.4.4 需求调研

产品立项后，则需要进行详细的需求调研。需求调研同样有两种模式：主动模式和被动模式。

在主动模式下，公司需要派出需求调研小组与客户直接沟通，以获得正确可靠的需求。被动模式是市场调研人员根据自己的经验判断，无明确的需求提出者，这样的项目有一定的风险。

需求调研是整个项目最为重要的环节，此环节输出的一切资料都是后期工作的依据。所以，很多公司在这个阶段会投入较长的时间，花费大量的人力、物力，以保证需求调研的充分及正确。从测试角度来说，测试人员需在这个阶段就参与进来，从初期的需求调研到后期的系统测试，测试人员能够在第一时间接触处初始需求，对整个项目质量是有积极意义的。

需求调研期间一般要做以下几件事：

- 1、与客户座谈，输出需求记录表；
- 2、走访实际业务流水线，掌握客户实际业务流程；
- 3、输出需求整理文档，最好做出 demo 让用户确认；
- 4、最终输出用户需求规格说明书；

需求调研中可能存在的问题

- 1、用户无法确切描述其所期望的系统功能；
- 2、无法获取实际的业务流程；
- 3、无法获得客户的支持，使得需求调研流于表面

在上述几个问题中，我们需要特别注意的是第一点，因为没有准确正确的需求，将会给我们的项目开发测试埋下炸弹，带来无尽的变更与返工。

- 需求调研参与人员：市场人员，开发人员，测试人员等

- 调研对象：客户或假想客户（广泛的用户群）
- 输出：需求规格说明书

### 1.4.5 设计开发

在输出需求规格说明书，经过评审确定后，则由项目组内的开发人员进行系统设计。一般的，如果公司有专门的系统架构师，则会由系统架构师进行系统的概要设计，在需求阶段已经提出系统的开发模式，在概要设计阶段，主要解决的是系统的整体架构，从稳定性，安全性，扩展性，可维护性等方面进行设计，然后再进行系统的详细设计，数据库设计等相关事宜。

- 设计人员：系统架构师、项目开发小组
- 输出：项目开发计划、概要设计文档、详细设计文档、数据库设计文档等

### 1.4.6 系统测试

在开发部门完成代码编写、模块集成后，由测试部门进行该项目的测试，按照前期的测试计划，利用测试用例进行系统的功能、性能测试。在经过多次版本的迭代后，完成系统测试，输出测试报告。

- 测试人员：项目测试小组
- 输出：测试计划、测试方案、测试用例、功能测试报告、性能测试报告等。

### 1.4.7 产品发布

经过开发部门、测试部门和其他部门的努力，产品在预定的日期完成，由项目组择日发布。

- 发布人员：项目实施人员、市场部等
- 输出：客户现场项目实施报告等

### 1.4.8 产品维护

交付使用后，需根据需求调研阶段的协议，制定产品维护流程，在出现问题需及时解决，直到产品使用废弃或升级，进行新的生命周期。

### 1.4.9 产品升级

在软件产品使用到一定期限后，可以根据先前的约定进行升级，或者根据客户新的需求，再次进行新需求的调研开发等。比如现在很多公司给一些企业做信息管理系统，类似于 ERP，往往会分成几期来做，这里面就有一个升级换代的问题。

其实，在这个过程中，也有产品是消亡的，比如奥运会的门票销售系统，在奥运会结束后，该系统的使用寿命也就结束了，主办方可能根据其他的需要进行定制，改版，应用到其他项目中去。

上面章节简单的描述了软件项目的一个运作的流程,整个过程与软件的生命周期是相对应的。在我们的软件测试活动中,一定要知道每个项目的由来,这样才能更好的开展我们的测试工作。

## 1.5 软件测试工作流程

一个企业或者公司的软件测试工作流程是什么样子的呢?没有工作经验或者不太了解软件测试工作的朋友可能没有一个清晰的概念,下面我将从实际角度出发,介绍一般软件公司中软件测试部门的工作开展流程。在这之前,我们先来看看测试部门的组织结构情况。

### 1.5.1 测试部门组织结构

从不同的角度出发,测试部门的构成可从这几个方面考虑:人员构成、技术构成、资源构成。实际上,一个测试部门的管理者,可从这些方面考虑部门的组织结构。

#### 人员构成

一个完整的测试部门,一般包括以下几个角色:测试主管、测试组长、环境保障人员、配置管理员、测试设计人员、测试工程师等等职位。如图 1-10 所示:

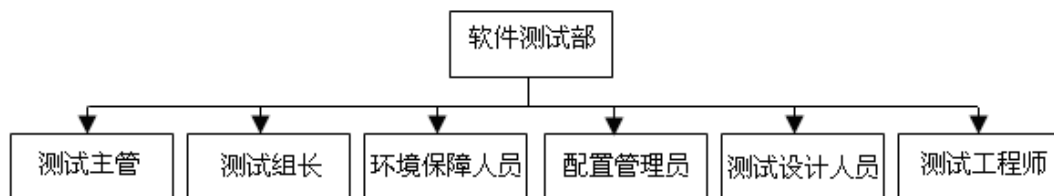


图 1-10 测试部组织结构图

#### 测试主管

测试主管负责测试部门的日常管理工作,负责部门的技术发展、工作规划等,同时他也是测试部门与其他部门的接口人,在其他兄弟部门需要测试部门协助或安排测试工作的时候,需要首先与测试主管沟通,提出申请。

#### 测试组长

测试组长隶属于测试部门,由测试主管指派。在接收到一个项目测试需求后,

测试主管会根据项目的实际情况，比如项目的技术要求、难易程度，指派合适的测试人员担当测试组长角色，由其负责该项目测试工作。有些公司称测试组长为测试经理。

### 环境保障人员

环境保障人员的作用是维护整个项目过程中的系统环境，如硬件、软件方面的。一般的公司不具备这样的人员，都由测试人员兼做，也可能有专职的保障人员，但不隶属于测试部门。所以该角色一般是重叠的。

### 配置管理员

配置管理是软件开发过程中一个极其重要的工作流程，在这个环境可以对需求变更、版本迭代、文档审核起到相当大的作用，所以稍微正规一些的公司都会配备配置管理员。

### 测试设计人员

一般由高级测试工程师担当，负责项目测试方法设计、测试用例设计以及功能测试、性能测试的步骤、流程设计。很多公司将该角色与测试工程师重叠，不严格区分测试设计人员与测试工程师角色。

### 测试工程师

测试工程师的实际工作内容大多数是执行测试用例，进行系统的功能测试，经过多次的版本迭代，完成系统测试。一般由初级测试工程师、中级测试工程师担当。

## 技术构成

技术构成主要是从测试部门需具备的技术角度来考虑，主要有这几类：白盒测试技术人员、黑盒测试技术人员、自动化测试技术人员、项目管理技术人员等等，如图 1-11 所示。

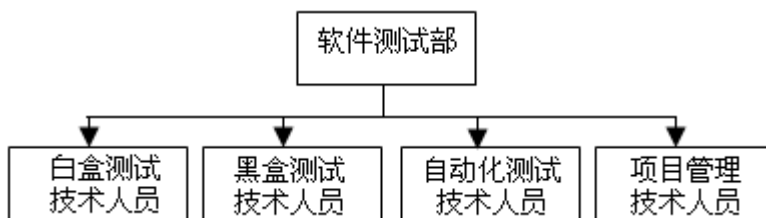


图 1-11 测试部技术构成图

## 白盒测试技术人员

该职位需要测试人员精通掌握软件的开发语言，一般需要有几年的开发经验，能够进行底层的代码 review、测试桩设计等，同时能够使用白盒测试工具对系统的最小功能单元进行测试，找出代码，系统架构方面的缺陷。

## 黑盒测试技术人员

黑盒测试技术人员一般要求测试人员具有一定软件工程理论，软件质量保证知识，需要从系统的功能实现，需求满足情况监察系统的质量，需要掌握基本的软件开发语言，数据库基本知识，操作系统基本知识，测试流程以及相应的工作经验。

## 自动化测试人员

自动化测试人员相对的要求较高，需要测试人员掌握软件开发的知識，系统的调优，自动化测试工具，如 QuickTestProfessional、LOADRUNNER 等，同时需要具备相当丰富的工作经验。目前国内这方面的人才比较缺。

## 项目管理技术人员

该角色要求项目管理人员掌握一般常用的项目管理知识，如配置管理，版本控制，评审管理，项目实施与进度控制等等，不一定具备多强的测试技术，但需要有丰富的项目管理经验以及沟通协调能カ。能够保证项目在一个可控的环境下稳定运作。

## 资源构成

资源构成主要考虑的是测试部门的组建需要哪些硬件、软件资源，主要包括这几个方面：硬件资源、软件资源、技术支持等，如 图 1- 12 所示。

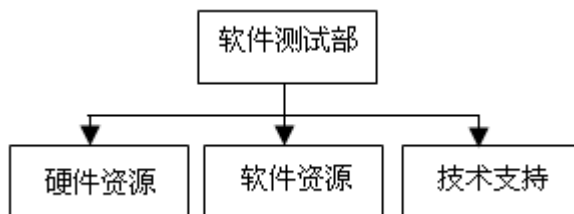


图 1- 12 测试部资源构成图

### 硬件资源

测试部门面临的是复杂多变的用户环境，需要具备在不同环境下进行系统的测试，所以就需要有齐备的测试环境，比如测试 PC 机，测试服务器，测试芯片，测

试手机等，需要模拟各式各样的用户环境，以保证在多变的环境下不会因为硬件的区别导致项目失败。

### 软件资源

在硬件具备的情况下，我们首先要考虑的是软件环境，如测试需要的操作系统、应用软件、管理软件等等。像我们平时使用的 WINDOWS，LINUX 等操作系统，SQL SERVER、ORACLE 等数据库软件，QuickTestProfessional、LOADRUNNER 等自动化测试工具，其实在当前的网络共享时代，这些都可以找到，不过需要注意版权问题。

### 技术支持

有时测试人员在遇到一些问题的时候，并不能靠自身的能力去解决，这需要兄弟部门给予支持，不管是技术还是其他方面的，确保在一个团队合作的环境下，更高效的完成测试工作，像华为公司在做测试项目时会指派对应的环境保障人员，或开发人员，作为技术支持。

## 1.5.2 测试工作流程

测试部门的工作流程严格意义上来说是按照软件的生命周期作为流转依据，主要有这几个环节：测试准备阶段、测试开展阶段、测试输出阶段等，如图 1-13 所示。



图 1-13 测试工作流程图

## 测试准备阶段

在一个项目开展的前期，需要进行需求调研等一系列的准备工作，这时测试部门需要做的是参与进前期的需求调研（但大多数不会），然后根据需求调研阶段生成的需求说明书指导下一步工作。在这个阶段，主要包括下面几个步骤。

### 测试计划制定

在项目立项后，项目经理会根据实际的情况，告知测试部主管，需要相应的测试小组参与进项目中来。那么测试主管会根据部门人员的构成以及技术构成进行协

调，指派一名测试组长。由测试组长负责该项目的测试工作。测试组长将会联系项目经理，获取项目的需求规格说明书，然后制定相应的测试计划，安排如何开展本项目的测试工作。具体流程可见 图 1- 14 所示。

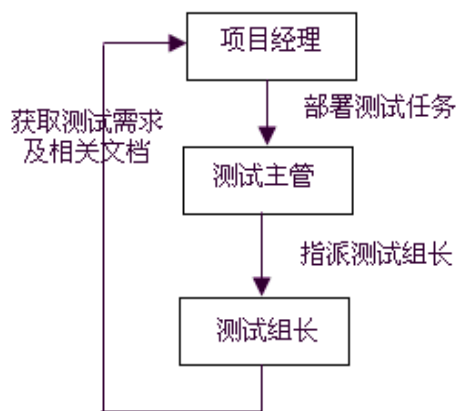


图 1- 14 测试工作介入流程图

案例：

项目经理张三告知测试主管李四，下周将有一个项目 A 需测试部门进行测试，已经提交了测试申请，请求测试经理安排人员测试。测试主管李四考察项目 A 状况，安排测试组长王五，由王五负责项目 A 测试工作，王五接到工作任务后，会从项目经理处获取项目需求以及相关的系统文档，然后制定该项目的测试计划。此时，测试部门将正式进入到该项目组。

### 测试小组建立

在测试组长制定了项目测试计划后，测试组长会根据项目周期长短，项目规模大小组建合适的测试小组，在组建测试小组过程中，多数情况下，小组成员多由测试主管指定。成立测试小组后，组长会召开测试项目的通气会，让组员清晰本项目的相关情况。图 1- 15 说明了测试小组的建立流程。

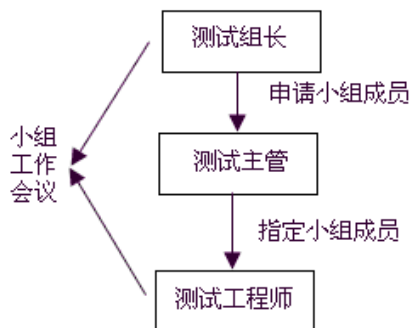


图 1-15 测试小组建立图

案例：

项目 A 测试组长王五，根据项目 A 的实际情况，向测试主管李四申请测试组员。测试主管李四，会根据目前部门的工作任务分配情况，指定合适的测试工程师参与进该测试小组。小组建立后，测试组长王五将召开小组会议，介绍当前项目的进展情况以及当前的工作任务。

### 需求测试启动

测试小组成立后，测试组长将会安排小组成员阅读需求文档及其他项目文档，开展需求测试工作。可以按照需求的功能结构划分测试任务，亦可整体阅读测试。此阶段，测试工程师需提交需求测试结果报告，并对测试结果报告进行评审，如果合格，则开展测试需求提取工作，如不合格，则由测试组长将需求测试结果反馈至需求文档及其他项目文档的相关提供部门进行校正，校正完成后再次测试，直至合格为止。图 1-16 显示了需求测试的流程图。

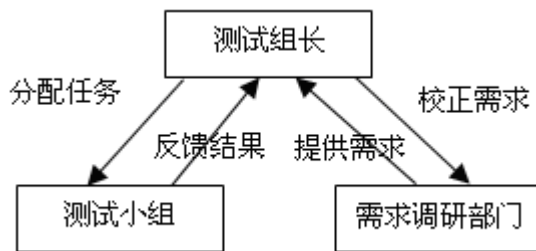


图 1-16 需求测试流程图

案例：

测试组长王五安排小组成员阅读项目 A 的需求规格说明书及其他项目文档，并开始需求测试，最后对测试结果进行评审，如果通过，则开始测试需求的提取，如果不通过，则由需求调研部门核实需求，修改后再次测试，直至通过为止。

### 测试需求提取

在需求评审通过后，测试组长将根据测试工程师的技术能力及工作经验，恰当的分配系统功能模块给他们，然后结合公司所使用的测试管理工具，比如 TestDirector，进行测试需求的提取，所谓测试需求，通俗点讲，就是我们需要测试的任务点。此阶段，按照正规的工作流程，仍需要进行评审活动，以检查需求提取过程中，是否在多余，遗漏等错误。评审合格后，进入测试用例编写流程。当然，在这个阶段也可能需要编写测试方案。图 1-17 展示了测试组长部署测试需求提取任务的一般流程。



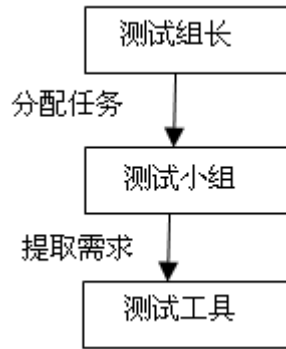


图 1-17 部署测试需求提取任务流程图

案例：

测试组长王五根据小组成员的工作能力，分配相应的系统测试模块，然后利用 TestDirector 进行测试需求的提取和管理。在经过多次评审通过后，将开始测试用例的编写。

### 测试用例编写

在测试需求提取工作完成后，我们就开始测试用例的编写。测试用例的编写是个重点和难点。测试用例是我们开展软件测试的指导性工件。在这个阶段，同样会有多次的测试用例评审会议，检查每个成员所写的测试用例的正确性及效率。同样，在这个阶段可以使用测试管理工具。图 1-18 是部署测试用例任务流程图

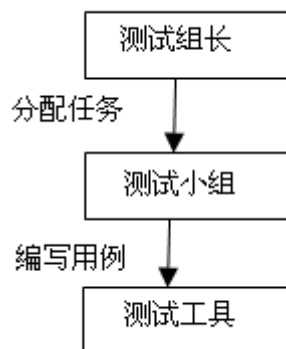


图 1-18 部署测试用例编写任务流程图

案例：

测试组员根据测试需求设计相应的测试用例，并利用 TestDirector 进行测试用例的管理。经过多次评审后，测试用例应该设计的比较完善了。此时，测试前的准备工作基本上就完成了。

## 测试开展阶段

测试准备阶段的任务完成及编码工作完成后,我们就可以开始正式项目测试工作了。这个过程比较简单,主要是搭建测试环境、文档引入及执行测试三个部分。

测试组长将负责搭建测试环境,当然也可以安排小组内其他人员搭建,此时,可根据需求规格说明书中的软件产品运行环境配置要求进行搭建,也可以从开发同事那里获取该软件的环境搭建单。这里需注意的,测试环境最好与开发环境分开。

接下来,需将本次测试过程中可能使用到的各种文档规划好,并告知小组成员如何使用这些文档,比如每天的工作日报,功能测试报告、性能测试报告等等相关文档的模板。

最后,就可以执行我们前期设计的测试用例了。根据项目的 Bug 管理流程,经过多次的版本迭代,完成测试工作。

## 测试输出阶段

测试工作开展过程中,需要输出很多工件,比如测试计划、测试方案、测试用例、测试工程师工作日报,功能测试报告,性能测试报告等等。这些都是软件测试过程中的输出工件。

项目经理会根据最终的软件产品测试报告,来衡量当前软件版本的质量,以决定是否发布。

## 本章练习

- 1、什么叫 C/S、B/S？
- 2、软件测试共有几种模型？具体内容是什么？相互之间有什么区别与联系？
- 3、简要描述同行评审与阶段评审的区别。
- 4、软件测试与软件开发的关系。
- 5、什么叫软件测试？软件测试的目的是什么？
- 6、软件测试的一般工作流程是什么？
- 7、软件测试的测试流程是什么？各阶段的工作内容重点是什么？
- 8、当你接到一个测试任务后，你如何开展测试工作？
- 9、压力测试、负载测试、性能测试、强度测试的含义各是什么？举例说明
- 10、什么叫冒烟测试？。

## 第 2 章 Windows Server 2003 服务配置

### 本章要点

本章重点讲解 Windows Server 2003 操作系统的安装以及 IIS 服务的安装配置，使读者对软件构成以及软件运行原理有个初步的认识。

### 2.1 Windows Server 2003 安装

Windows Server 2003 作为专业的 Windows 服务器产品，其主要优点表现在如下几个方面：

#### 可靠

Windows Server 2003 是迄今为止提供的最快、最可靠和最安全的 Windows 服务器操作系统。目前最新的 Windows 服务器系统是 Windows 2008，但尚未普及。Windows Server 2003 提供集成结构，用于确保商务信息的安全性；提供可靠性、可用性和可伸缩性，提供用户需要的网络结构。

#### 高效

Windows Server 2003 提供各种工具，允许用户部署、管理和使用网络结构以获得最大效率。Windows Server 2003 提供灵活易用的工具，有助于使用户的设计和部署与单位和网络的要求相匹配；通过加强策略、使任务自动化以及简化升级来帮助用户主动管理网络；通过让用户自行处理更多的任务来降低支持开销。

#### 联网

连接 Windows Server 2003 可以帮助用户创建业务解决方案结构，以便与雇员、合作伙伴、系统和用户更好地沟通。Windows Server 2003 提供集成的 Web 服务器和流媒体服务器，帮助用户快速、轻松和安全地创建动态 Intranet 和 InternetWeb 站点；提供集成的应用程序服务器，帮助用户轻松地开发、部署和管理 XML Web 服务；提供多种工具，使用户得以将 XML Web 服务与内部应用程序、供应商和合作伙伴连接起来。

## 经济

与来自微软公司的许多硬件、软件 and 渠道合作伙伴的产品和服务相结合，Windows Server 2003 提供了有助于使用户的结构投资获得最大回报的选择。Windows Server 2003 为使用户得以快速将技术投入使用的完整解决方案提供简单易用的说明性指南；通过利用最新的硬件、软件和方法来优化服务器部署，从而帮助用户合并各个服务器；降低用户的所属权总成本（TCO），使投资很快就能获得回报。

本课程使用在虚拟机中安装 Windows Server 2003 Enterprise Edition 的方式，目的让学者掌握虚拟机的使用方法以及如何在虚拟机中操作系统的安装方法。

约定：

Windows Server 2003 版本为 Windows Server 2003 Enterprise Edition SP1

虚拟机版本：VMware Workstation 5.5.3 中文汉化版

在使用虚拟机之前，我们先来了解一下虚拟机的相关知识。

## 虚拟机原理

通过软件模拟、具有完整硬件系统功能、运行在一个完全隔离环境中的完整计算机系统。通过虚拟机软件，我们可以在一台物理计算机上模拟出一台或多台虚拟的计算机，这些虚拟机就像真正的计算机那样进行工作。例如我们可以在虚拟机上安装操作系统、安装应用程序、访问网络资源等等。对于用户而言，它只是运行在你物理计算机上的一个应用程序，但是对于在虚拟机中运行的应用程序而言，它就像是在真正的计算机中进行工作。

因此，当我们在虚拟机中进行学习或者工作时，可能系统一样会崩溃，但是崩溃的只是虚拟机上的操作系统，而不是物理计算机上的操作系统，并且我们可以使用虚拟机的“Undo”（恢复）功能，马上恢复虚拟机到安装软件之前的状态。

虚拟机是一个抽象的计算机，和实际的计算机一样，具有一个指令集并使用不同的存储区域。它负责执行指令，还要管理数据、内存和寄存器。这台虚拟的机器在任何平台上都提供给编译程序一个的共同的接口。编译程序只需要面向虚拟机，生成虚拟机能够理解的代码，然后由解释器来将虚拟机代码转换为特定系统的机器码执行。

## 虚拟机优点

在日常工作学习过程中，如果要在一台电脑上装多个操作系统，不用虚拟机的话，有两个办法：一是装多个硬盘，每个硬盘装一个操作系统，这个办法代价比较昂贵；二是在一个硬盘上装多个操作系统，这个办法不够安全，因为硬盘 MBR 是操作系统的必争之地，搞不好会几个操作系统同归于尽，而使用虚拟机软件既省钱又安全，对想学 linux 和 unix 的朋友来说很方便。虚拟机可以在一台机器上同时运行几个操作系统，既节省了硬件资源，又提供很大的便利。

在我们的测试工作中，如果需要做兼容性的测试，需要安装多个版本的操作系统，我们则可在虚拟机上完成。同样，对于一些企业来讲，他们所有的服务器都是配置较为强劲，然而大多数这些服务器的使用率较为低下，比如可能只用了服务器系统资源的 40%，这样就会造成一定的浪费，如果在这些服务器上安装虚拟机，则提供更多的可用系统，最大程度的减少硬件资源的浪费。

## 虚拟机缺点

虚拟机的安装使用，受系统硬件资源的影响比较大。一般的，如果机器的硬件资源不足，比如内存，硬盘，CPU 配置较差的话，运行虚拟机则较为吃力。同时，虚拟机在软件测试工作的应用，个人觉得仅限于功能性的测试工作，而不使用性能测试。

在了解了虚拟机相关知识后，我们开始学习如何在虚拟机上安装 Windows Server 2003。

### 2.1.1 虚拟机安装

虚拟机安装很简单，这里的安装程序是在多特软件下载的破解版本，这里不做详细描述。安装时根据界面提示即可。

虚拟机安装完成后，会在桌面生成名为“VMware Workstation”快捷方式，点击该图标，打开虚拟机软件。

### 2.1.2 Windows Server 2003 安装

启动虚拟机后，按照下列步骤进行 Windows Server 2003 系统安装。

1、点击“新建虚拟机”，进入新建虚拟机向导，如图 2-1 所示。

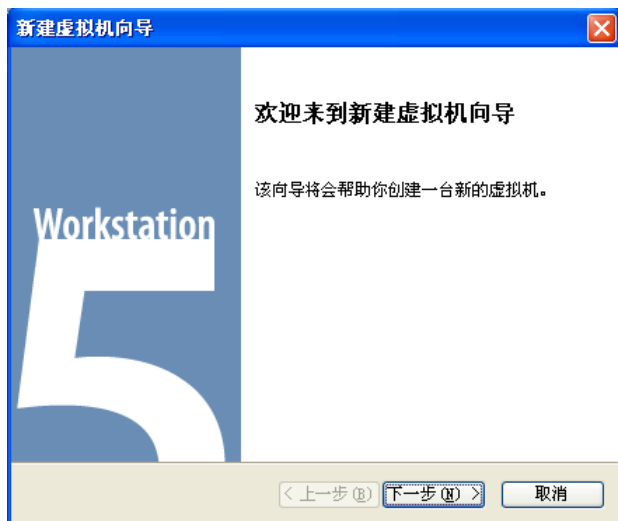


图 2-1 新建虚拟机向导

2、选择“典型”设置，点击【下一步】，如图 2-2 所示。

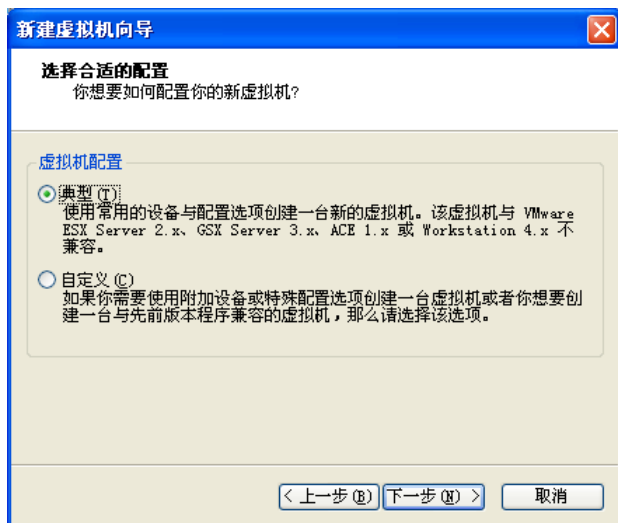


图 2-2 虚拟机配置

3、按图示选择客户机操作系统与版本，点击【下一步】，如图 2-3 所示。



图 2-3 选择客户机操作系统

4、设置虚拟机名称、修改虚拟机文件存放位置，点击【下一步】，注意：存放位置所在分区须至少有 10G 以上可用空间。



图 2-4 设置虚拟机名称

5、设置虚拟机网络连接方式，这里使用桥接方式，点击【下一步】。



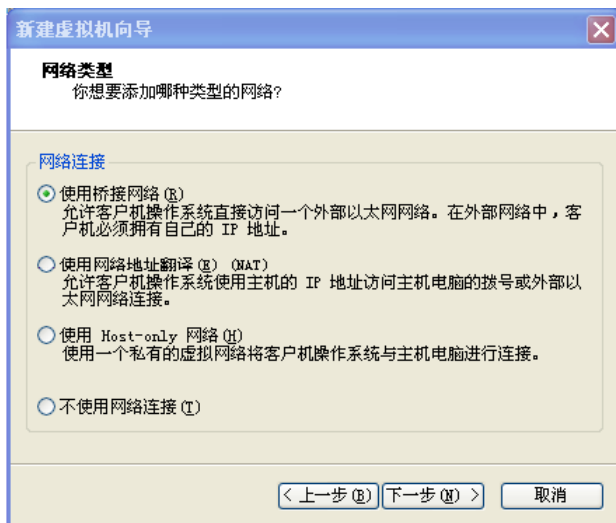


图 2-5 配置虚拟机网络

6、设置虚拟机硬盘空间，默认即可，点击【完成】。

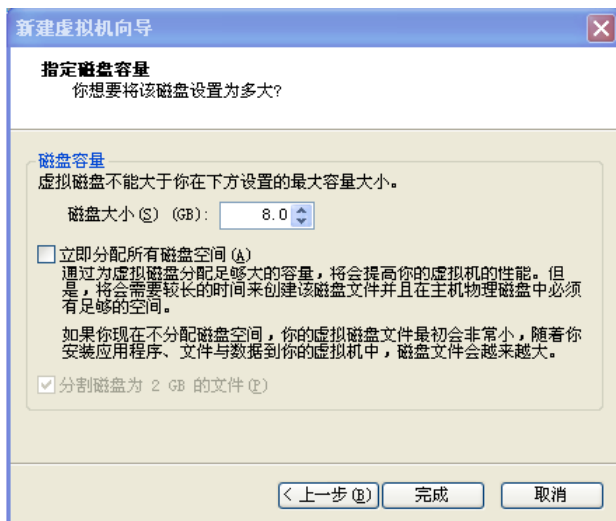


图 2-6 指定虚拟机磁盘容量

7、点击“编辑虚拟机设置”。

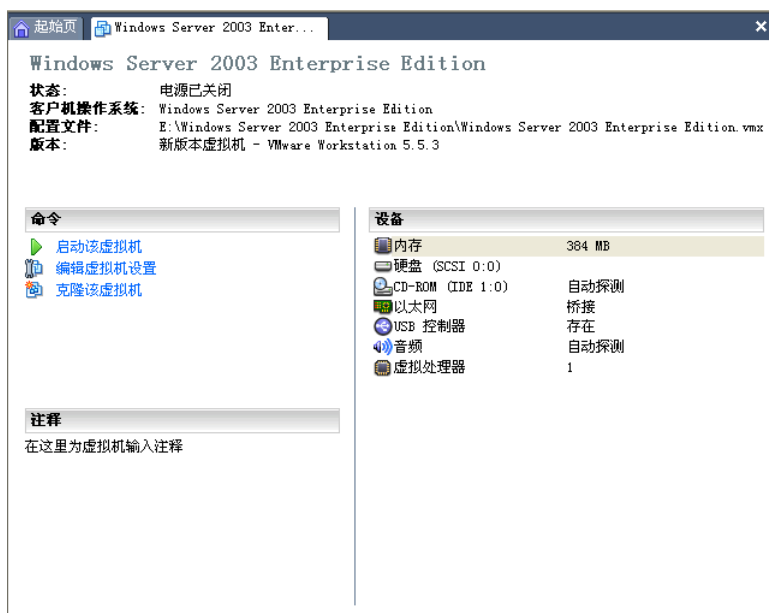


图 2-7 编辑虚拟机设置

8、设置CD-ROM连接方式为“使用ISO镜像”，并设置ISO镜像文件，如图 2-8 所示。

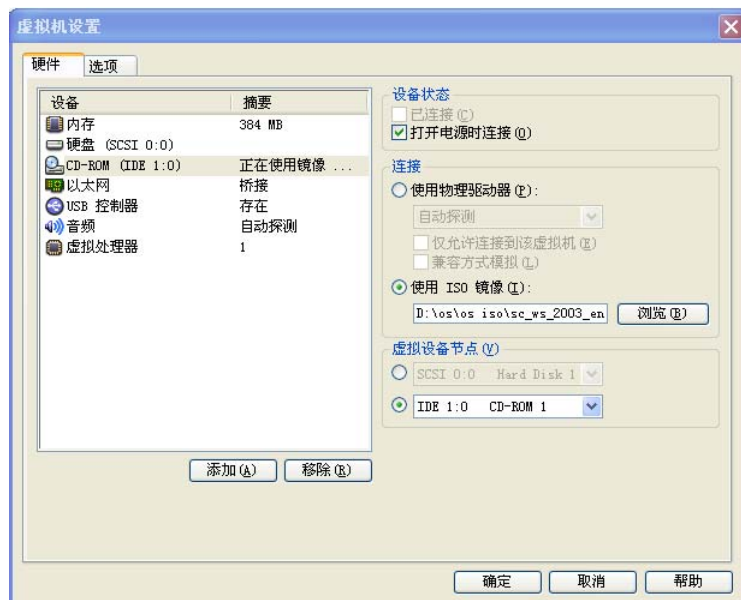


图 2-8 设置光驱连接方式

9、点击“启动该虚拟机”，开始安装。

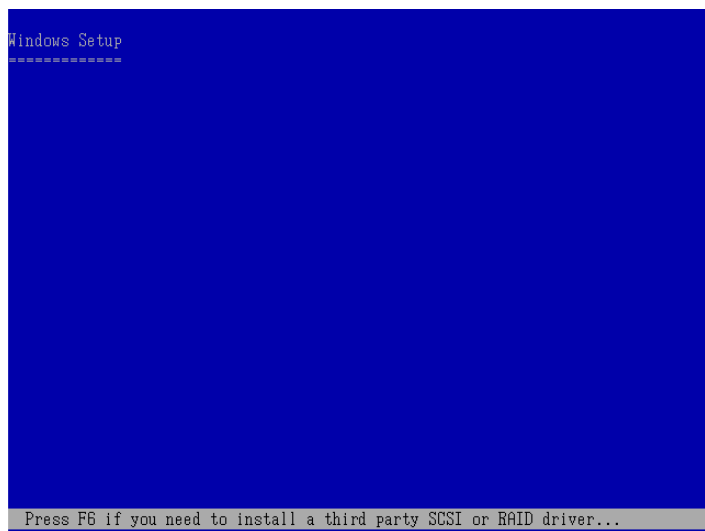


图 2- 9 Windows Server 2003 安装界面

10、回车，不分区，直接安装。

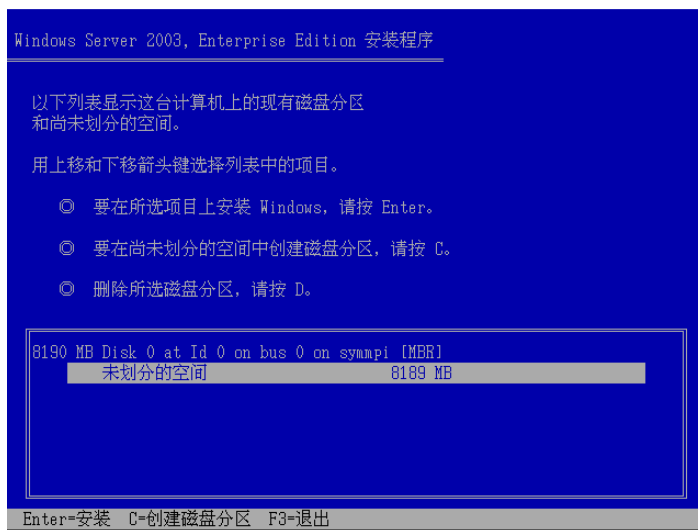


图 2- 10 创建磁盘分区

11、按图示选择，回车格式化虚拟硬盘分区，继续安装。

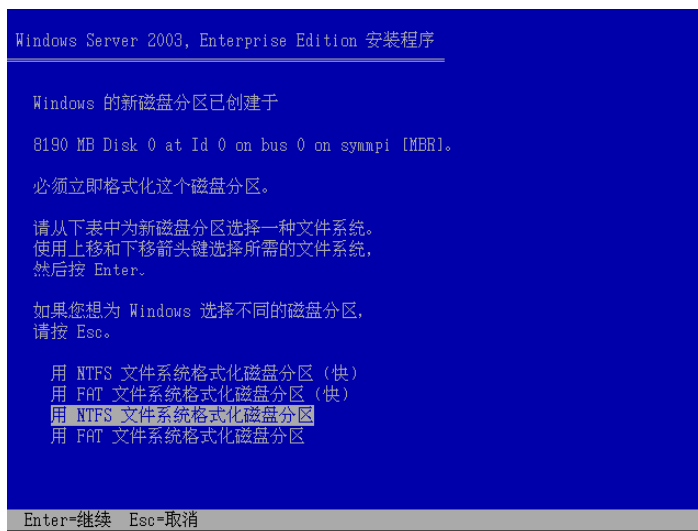


图 2- 11 选择文件系统格式化磁盘

## 12、格式化过程。



图 2- 12 格式化过程

## 13、安装文件复制过程。

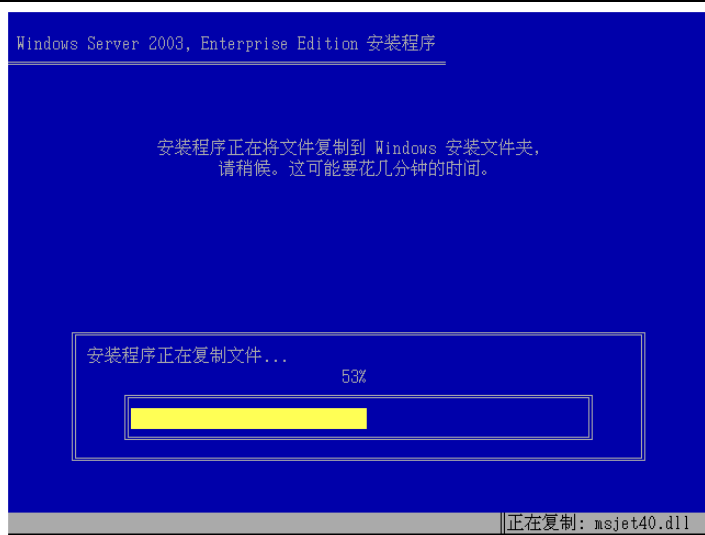


图 2-13 文件复制进程

14、重启后，进入安装界面。



图 2-14 系统安装进程

15、安装过程。

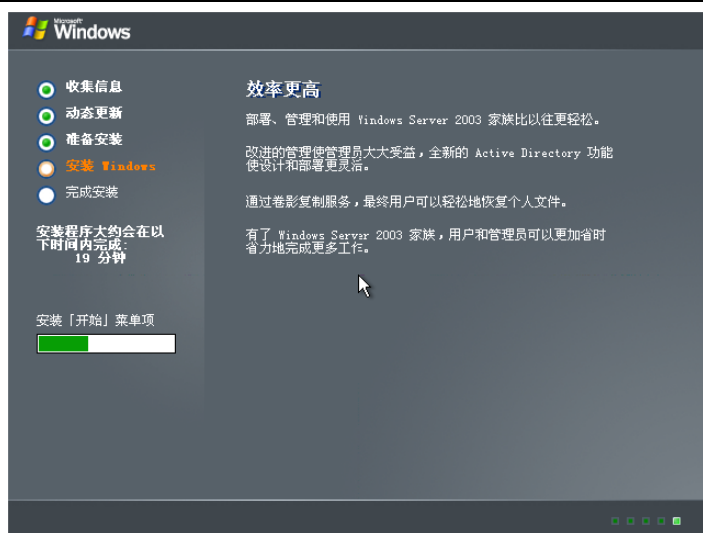


图 2- 15 系统安装进程

16、安装完成，Windows Server 2003 系统界面如图 2-16 所示。



图 2- 16 Windows Server 2003 系统界面

因为使用是的整合版，所以安装过程比较简单，没有需要我们手动输入信息的地方。如果是其他安装版本，可能需我们手动输入相关信息，比如许可证、一些配置信息等，大家根据实际情况输入即可。

安装好的虚拟系统与我们局域网的其他机器一样，从某种意义上讲，也就是一台真实存在的电脑，所以虚拟系统的使用与平时大家使用的系统一样，比如文件共享，上网等操作。

小提示:

这里使用的是美化版 Windows Server 2003 系统, 故界面与普通的 Windows Server 2003 系统界面有些不一样。

## 2.2 Windows Server 2003 部署 IIS 服务

IIS 是 Internet Information Server 的缩写, 它是微软公司主推的服务器。目前主流的版本是 Windows 2003 里面包含的 IIS 6.0, 新的 7.0 版本尚未普及。IIS 与 WindowNT Server 完全集成在一起, 因而用户能够利用 Windows NT Server 和 NTFS (NT File System, NT 的文件系统) 内置的安全特性, 建立强大, 灵活而安全的 Internet 和 Intranet 站点。IIS 支持 HTTP (Hypertext Transfer Protocol, 超文本传输协议), FTP (Fele Transfer Protocol, 文件传输协议) 以及 SMTP 协议, 通过使用 CGI 和 ISAPI, IIS 可以得到高度的扩展。IIS 支持与语言无关的脚本编写和组件, 通过 IIS, 开发人员就可以开发新一代动态的, 富有魅力的 Web 站点。IIS 不需要开发人员学习新的脚本语言或者编译应用程序, IIS 完全支持 VBScript, JScript 开发软件以及 Java, 它也支持 CGI 和 WinCGI, 以及 ISAPI 扩展和过滤器。IIS 的设计目的是建立一套集成的服务器服务, 用以支持 HTTP, FTP 和 SMTP, 它能够提供快速且集成了现有产品, 同时可扩展的 Internet 服务器。

IIS 相应性极高, 同时系统资源的消耗也是最少, IIS 的安装, 管理和配置都相当简单, 这是因为 IIS 与 Windows NT Server 网络操作系统紧密的集成在一起, 另外, IIS 还使用与 Windows NT Server 相同的 SAM (Security Accounts Manager, 安全性账号管理器), 对于管理员来说, IIS 使用诸如 Performance Monitor 和 SNMP (Simple Network Management Protocol, 简单网络管理协议) 之类的 NT 已有管理工具。IIS 支持 ISAPI, 使用 ISAPI 可以扩展服务器功能, 而使用 ISAPI 过滤器可以预先处理和事后处理储存在 IIS 上的数据。用于 32 位 Windows 应用程序的 Internet 扩展可以把 FTP, SMTP 和 HTTP 协议置于容易使用且任务集中的界面中, 这些界面将 Internet 应用程序的使用大大简化, IIS 也支持 MIME (Multipurpose Internet Mail Extensions, 多用于 Internet 邮件扩展), 它可以为 Internet 应用程序的访问提供一个简单的注册项。

IIS6.0 相比 IIS5.0 有了重大的提高和改进, 具有很多优秀的特性:

1、应用程序池: IIS6 可以将单个的 Web 应用程序或多个站点分隔到一个独立的进程 (称为应用程序池)。应用程序池以独立进程的方式极大的提高了 Web 服务器的安全性和稳定性。该进程与操作系统内核直接通信。当在服务器上提供更多的活动空间时, 此功能将增加吞吐量和应用程序的容量, 从而有效地降低硬件需求。

这些独立的应用程序池将阻止某个应用程序或站点破坏服务器上的 XML Web 服务或其他 Web 应用程序。

2、IIS6.0 还提供状态监视功能以发现、恢复和防止 Web 应用程序故障。在 Windows Server 2003 上, Microsoft ASP.NET 本地使用新的 IIS 进程模型。这些高级应用程序状态和检测功能也可用于现有的在 Internet Information Server 4.0 和 IIS 5.0 下运行的应用程序, 其中大多数应用程序不需要任何修改。

3、集成的 .NET 框架(DOTNET) Microsoft .NET 框架是用于生成、部署和运行 Web 应用程序、智能客户应用程序和 XML Web 服务的 Microsoft .NET 连接的软件和技术编程模型, 这些应用程序和服务使用标准协议(例如 SOAP、XML 和 HTTP) 在网络上以编程的方式公开它们的功能。 .NET 框架为将现有的投资与新一代应用程序和服务集成起来而提供了高效率的基于标准的环境

4、连接并发数,网络流量等监控, 这样可以使不同网站完全独立开, 不会因为某一个网站的问题而影响到其他网站。

5、IIS6.0 提供了更好的安全性 通过将运行用户和系统用户分离的方式. IIS 服务运行权限和 Web 应用程序权限 分开,保证 web 应用的足够安全.这些是其他 Web 服务器 所欠缺的。采用 IIS6.0 作为 Web 服务器, 确保我们的网站足够安全, 稳定,快速。

在了解了什么是 IIS 后, 我们开始学习在 Windows Server 2003 中配置 IIS 服务。首先, 确保有正常运行的 Windows Server 2003, 如果没有, 可参考“2.1.2 Windows Server 2003 安装”中的安装方法自行安装。

### 2.2.1 安装 IIS 服务

1、登录到 Windows Server 2003, 打开 IE, 输入 <http://localhost>, 如果出现 图 2-17 所示界面, 则表明 IIS 服务已经安装。



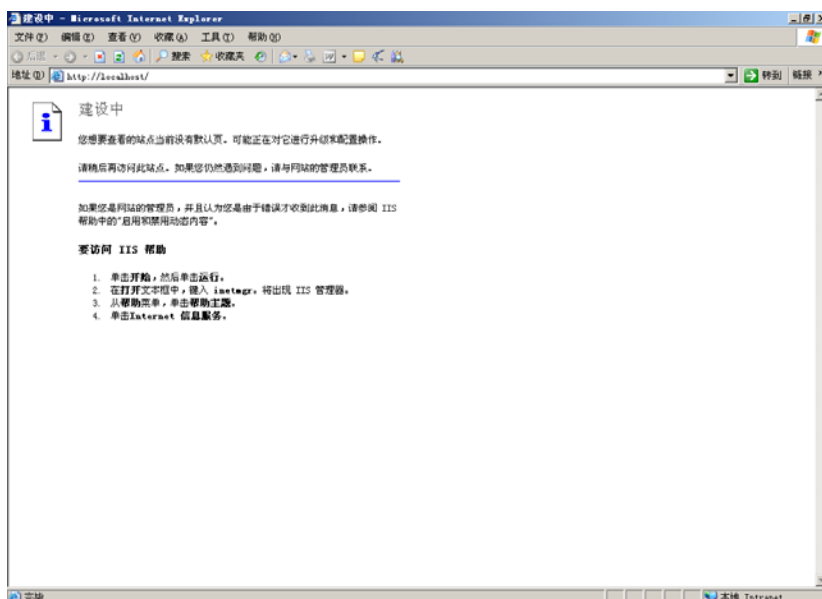


图 2-17 服务已安装状态图

2、如果出现 图 2-18 所示信息，则说明 IIS 没有安装，需按照下列步骤进行安装。

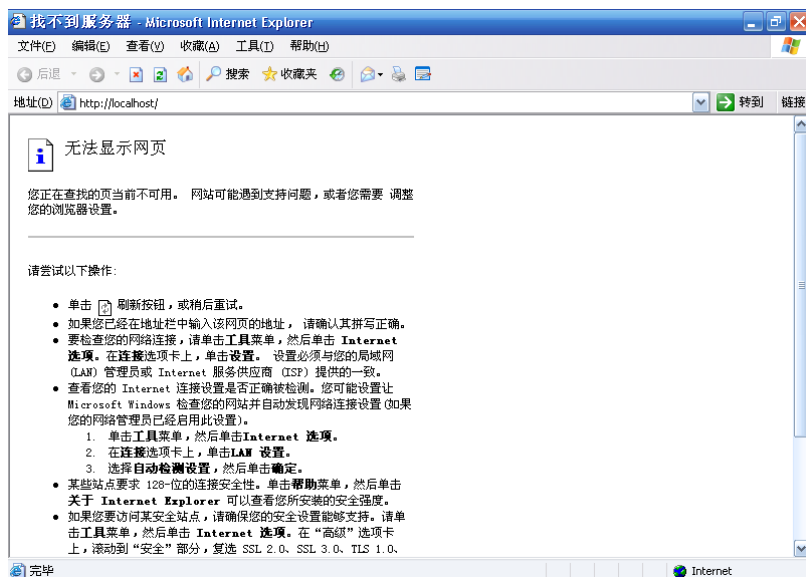


图 2-18 IIS 服务未安装状态图

3、打开控制面板->添加删除程序，如 图 2-19 所示。



图 2-19 添加或删除程序界面

4、点击“添加/删除 Windows 组件 (A)”，出现图 2-20。



图 2-20 Windows 组件

5、向下拖动右边滚动条，选中“应用程序服务器”，点击 **详细信息(I)...**，出现图 2-21。

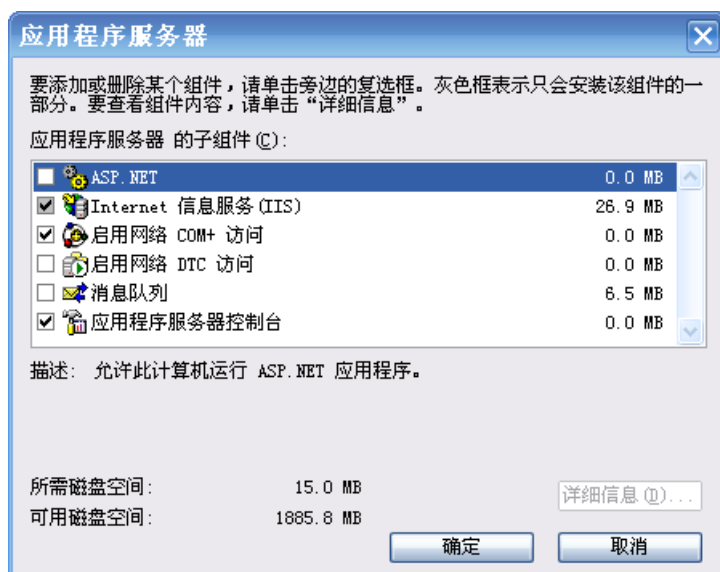


图 2-21 应用程序组件

6、勾选 ASP.NET，点击【OK】，返回到 Windows 组件向导界面，点击【下一步】，进行 IIS 服务安装。如果安装时需要插入 Windows Server 2003 系统光盘，则插入相应的系统光盘即可。点击【确定】，继续安装，如图 2-22 所示。

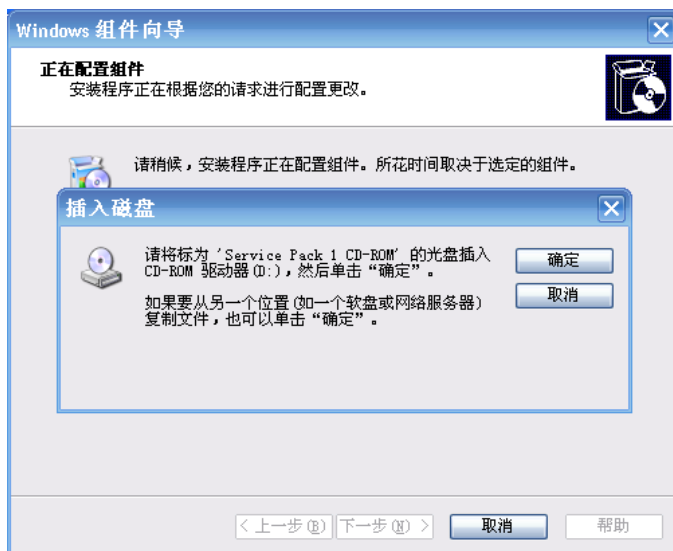


图 2-22 提示插入磁盘

7、安装过程显示。

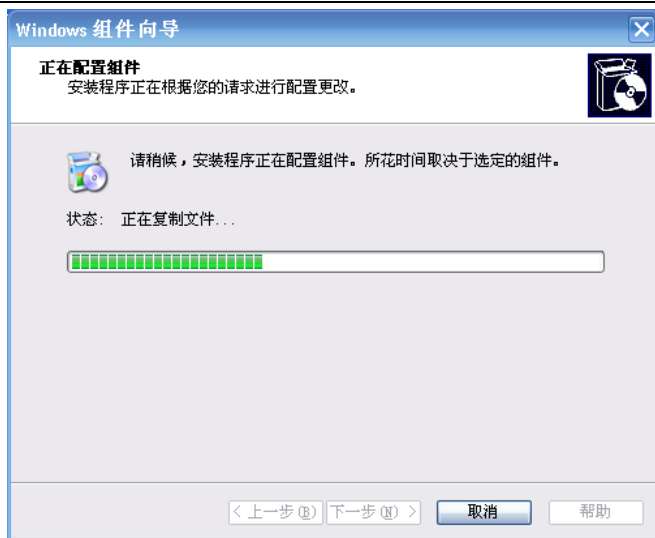


图 2-23 IIS 安装进程

8、安装完成后，点击【完成】，即可完成IIS服务的安装，如 图 2-24。安装后，最好注销一下机器。

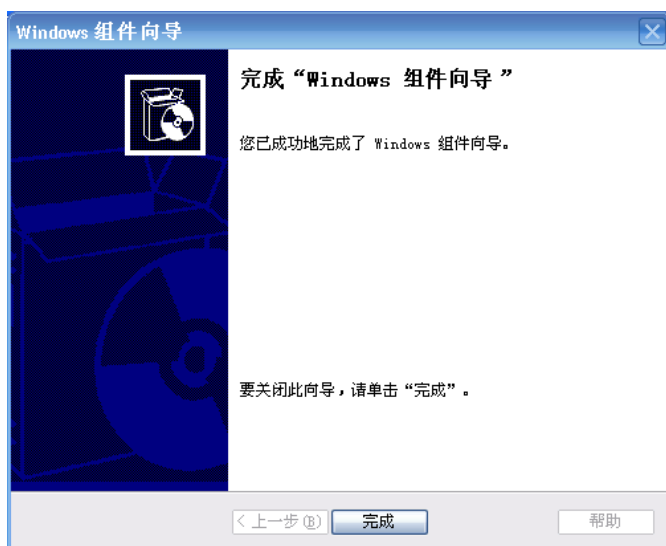


图 2-24 IIS 完成安装

9、打开 IE，输入 <http://localhost/>，如果出现“建设中”，则表示 IIS 服务安装成功。

### 2.2.2 部署应用程序

正确安装 IIS 服务后，我们就需要部署应用程序了。

## 配置 IIS 服务

1、打开控制面板->管理工具-> Internet 信息服务(IIS)管理器，如图 2- 25 所示。

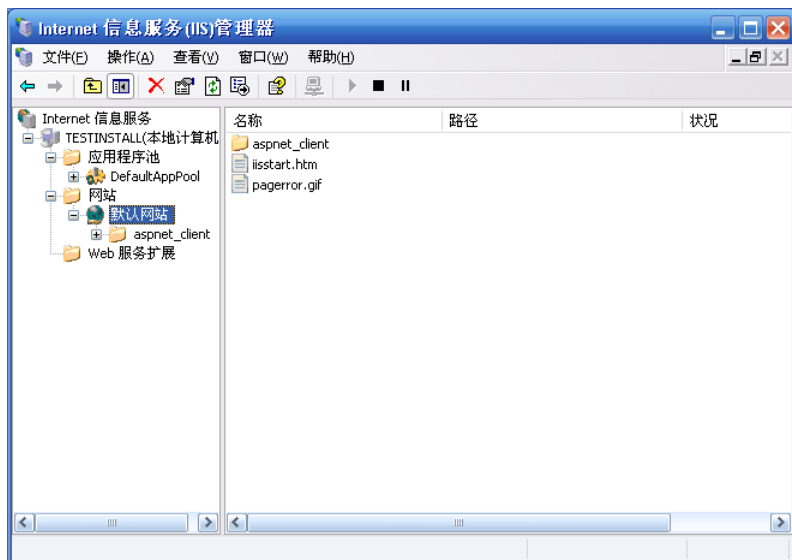


图 2- 25 Internet 信息服务管理器

2、点击“网站”，选中“默认网站”，右键选择“属性”，如图 2- 26 所示。



图 2- 26 默认网站属性-网站

下面简要描述该界面上几个重要功能界面。

- 网站：描述当前站点的一些信息，比如网站名称，端口、连接超时设置等。
- 网站标识：当前站点属性的设置，如网站名称、IP 地址、端口。默认情况下，这里不用修改。如果系统的 80 被占用时，可将此处的端口进行修改。
- 连接：用户访问网站时，如果在一段时间内请求没有被响应，则被认为是访问超时，这里可设置超时时间，默认为 120 秒。保持 HTTP 连接，表示虽然超时，但连接未断，请求被响应，立刻返回结果。一般默认，不做修改。
- 启用日志记录：IIS 服务的运行日志，一般默认，不做修改。



图 2-27 默认网站属性-主目录

- 主目录：该模块是 IIS 服务配置的核心部分。主要提供了应用程序部署路径，访问权限、应用程序设置等方面的功能。
- 路径及权限控制：本地路径是 IIS 服务中部署应用程序的路径，系统默认为 c:\inetpub\wwwroot，将我们需要运行的应用程序放在该目录下，IIS 服务就能解析应用程序中的相关请求，实现程序的功能。访问权限限制了用户访问应用程序的活动范围，一般默认为只读，不可写入。但对于大多数的应用系统来说，写权限是必须的，所以我们需在这里勾选“写入”。
- 配置：配置是为应用程序所做的一些设置。点击“配置”，出现 图 2-28。



图 2-28 应用程序配置

3、一般情况下，我们对这里不做什么设置，默认即可，但选项中的某些地方需要修改。点击“选项”标签，出现图 2-29。



图 2-29 应用程序配置-选项

4、将上图中“启用父路径”勾中，点击【确定】完成应用程序的配置。

5、点击“Web服务扩展”，右边出现图 2-30。



图 2- 30 Web 服务扩展

6、选中“Active Server Pages”，然后单击“允许”，启用 IIS 服务处理 ASP 语言功能。

到此，IIS 服务的一些简单配置就完成了。接下来是进行应用程序的部署。我们这里以《在线考试系统》为例。

### 部署应用程序

#### 放置应用程序

1、将需部署的应用程序放在一个文件夹里，如图 2- 31 所示。

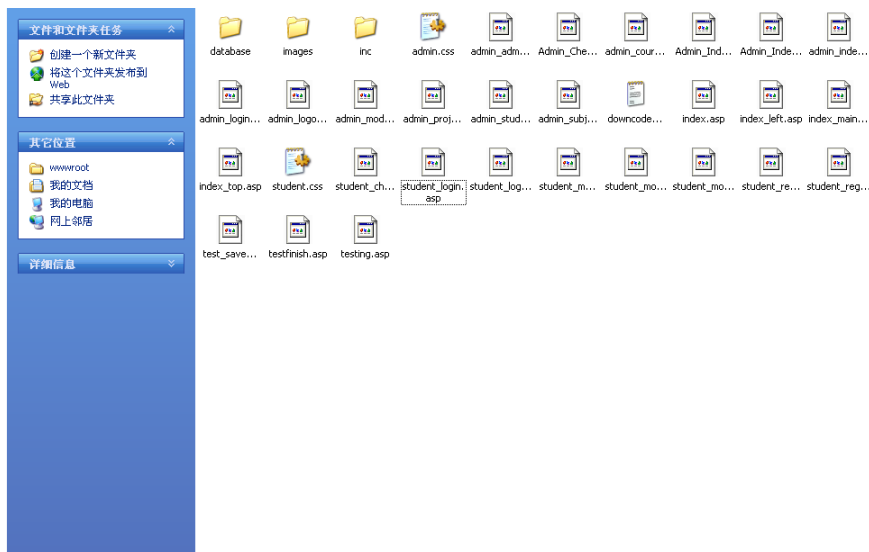




图 2-31 应用程序文件列表

2、将这些程序文件放入名为 exam 的文件夹，这里需提醒的是，一般情况下，我们不需要将程序文件放在太深的目录结构中。

3、拷贝 exam 文件，粘贴到 IIS 服务的默认网站路径 c:\inetpub\wwwroot 下。

### 修改应用程序执行权限

1、打开控制面板->管理工具-> Internet 信息服务(IIS)管理器，如图 2-32 所示。

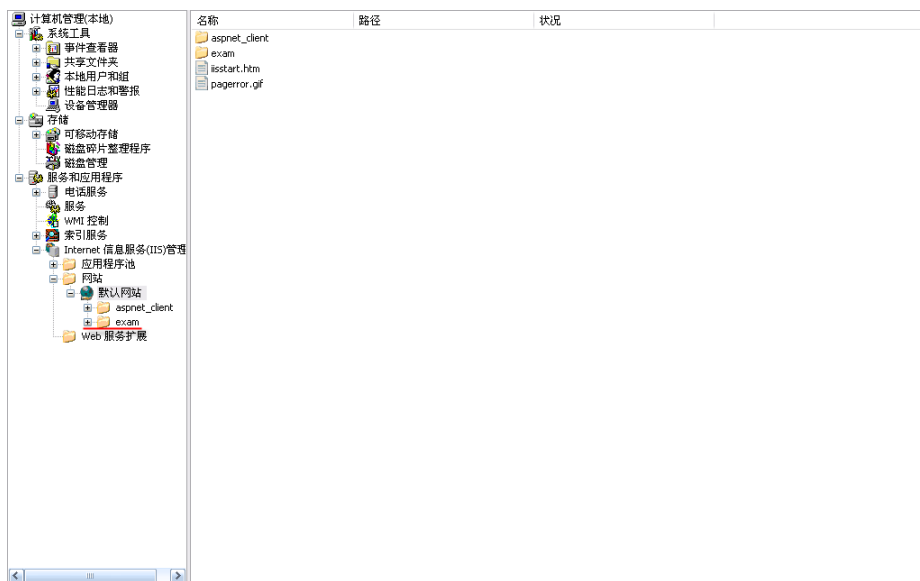


图 2-32 Internet 信息服务(IIS)管理器

2、可以看到图中红色标志处，已经出现了我们刚才部署的在线考试系统。选中“exam”，点击右键，选择“属性”，出现图 2-33。



图 2-33 应用程序 exam 属性

3、将执行权限处修改为“纯脚本”，修改完成后点击【确定】。完成 ASP 应用程序的部署。

4、打开 IE，输入 `http://localhost/exam/index.asp`，验证部署是否成功，成功界面如图 2-34 所示。

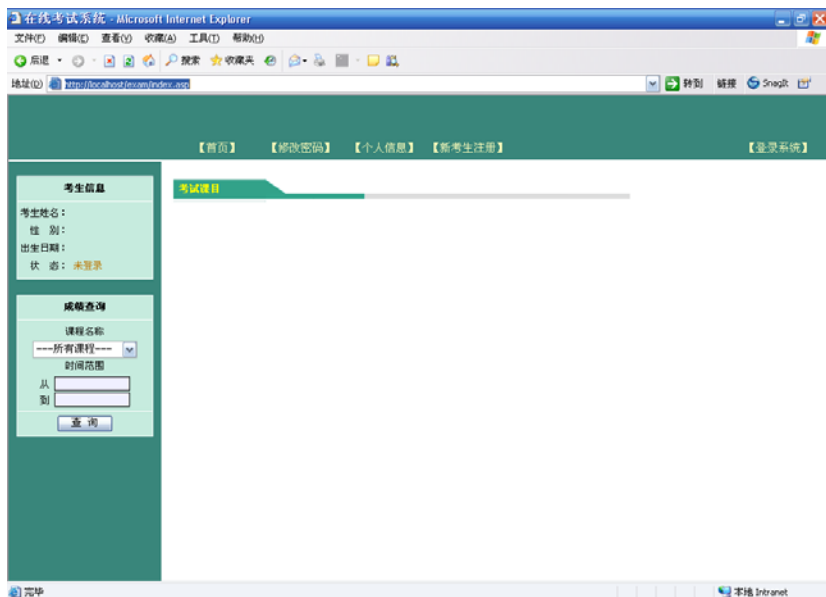


图 2-34 exam 部署成功

## 配置数据源

配置数据源这一步一般来说不用，只有在系统找不到数据库，无法正常使用时，可以手动添加数据源，以保证系统能正常工作。

1、打开控制面板->管理工具->数据源 (ODBC),如 图 2- 35 所示。

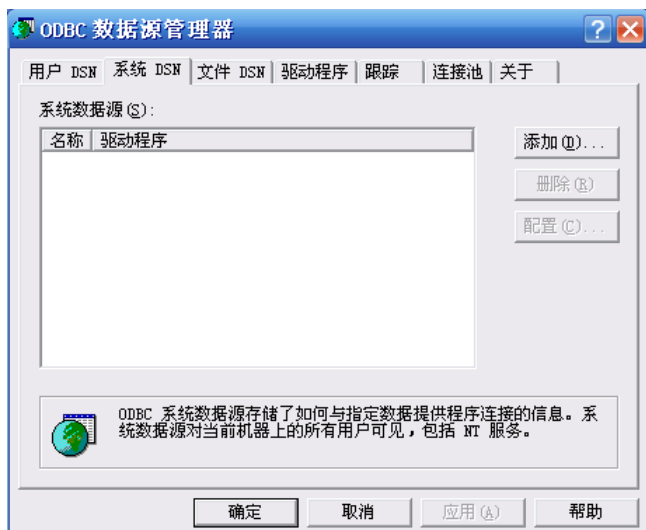


图 2- 35 ODBC 数据源管理器

2、切换页签到“系统DSN”，点击【添加】，出现 图 2- 36。



图 2- 36 选择数据源驱动程序

3、根据应用程序所使用数据库类型选择对应的驱动程序，比如我们这里的《在线考试系统》采用的是ACCESS数据库，那么就选择“Microsoft Access Driver (\*.mdb)”即可。如果是其他数据库，那么选择相应的驱动即可。选好点击【完成】，出现 图 2- 37。

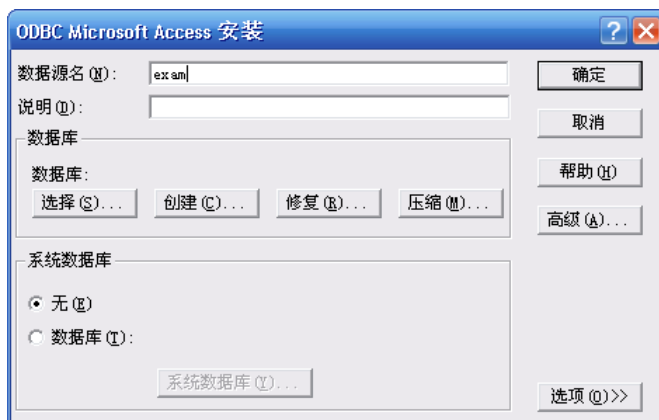


图 2-37 配置数据源

4、在数据源名中输入数据源的名称，如此处的“exam”，点击数据库中的【选择(S)】，出现图 2-38，进行数据库文件选取。



图 2-38 选择数据库文件

5、目录中是应用程序的数据库存放位置，如此处《在线考试系统》的数据库存放路径为 C:\Inetpub\wwwroot\exam\database，所以，我们在目录处进入到该目录，选中 exam.mdb，然后点击【确定】，完成数据库选择操作。

**小提示：**

如果我们的被测系统使用的是其他的数据库，比如 SQL Server、MySQL，那么需要在被测软件的数据库连接文件中修改连接信息。

6、一路确定，完成数据源的设置。重新打开 IE，访问被测系统，验证能否正常使用。如有问题，需根据问题描述再做检查。

### 2.2.3 调试运行 IIS 服务

在上述的配置部署过程中，可能出现各种各样的问题，下面列出几种常出的错误，以及解决办法。

#### 1、HTTP 错误 404 - 文件或目录未找到。

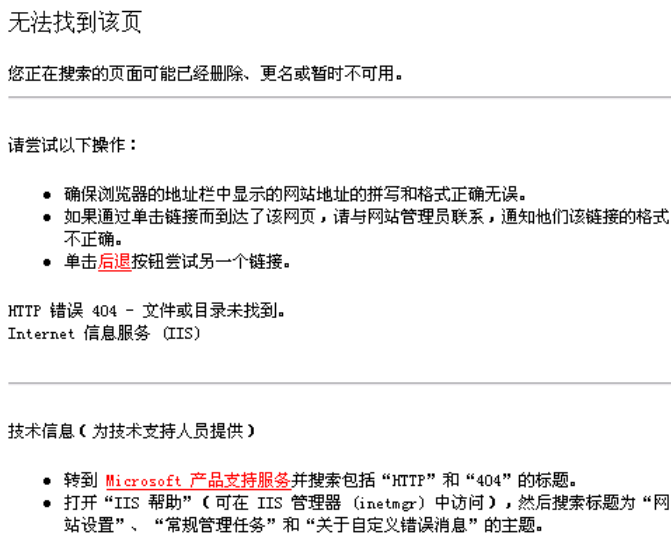


图 2-39 IIS 服务错误图-HTTP 错误 404

原因分析：上图出现的原因可能是因为 IIS 服务没有开启解析 ASP 程序功能，故提示文件或目录未找到。

解决办法：启用 IIS 服务解析 ASP 程序功能。

#### 2、HTTP 错误 403.1 - 禁止访问：执行访问被拒绝。

该页无法显示

您试图从目录中执行 CGI、ISAPI 或其他可执行程序，但该目录不允许执行程序。

请尝试以下操作：

- 如果您认为该目录应该允许执行访问权限，请与网站管理员联系。

HTTP 错误 403.1 - 禁止访问：执行访问被拒绝。  
Internet 信息服务 (IIS)

技术信息（为技术支持人员提供）

- 转到 [Microsoft 产品支持服务](#) 并搜索包括“HTTP”和“403”的标题。
- 打开“IIS 帮助”（可在 IIS 管理器 (inetmgr) 中访问），然后搜索标题为“配置 ISAPI 扩展”、“配置 CGI 应用程序”、“使用网站权限保护站点”和“关于自定义错误消息”的主题。
- 在 IIS 软件开发工具包 (SDK) 或 [MSDN Online Library](#) 中，搜索标题为“Developing ISAPI Extensions”、“ISAPI and CGI”和“Debugging ISAPI Extensions and Filters”的主题。

图 2-40 IIS 服务错误图-HTTP 错误 403

原因分析：上图问题的原因是没有执行 ASP 程序的权限，需要给用户执行该目录的权限。

解决办法：修改应用程序执行权限。

### 3、无法完成数据添加操作

如果在进行数据操作时，比如注册，登录等。出现

Microsoft JET Database Engine 错误 '80040e09'

不能更新。数据库或对象为只读。

/exam/student\_regpost.asp, 行 68

图 2-41 ASP 数据写入错误

原因分析：无法写入数据，虽然在 IIS 服务已经赋予了读写权限，但存放应用程序的磁盘分区格式可能是 NTFS 的，这样还需要对该应用程序文件夹赋予相应的权限，如给 exam 文件夹赋予 everyone（所有用户）读写权限。

解决办法：选择 exam 文件夹，右键查看属性，点击“安全”，如图 2-42 所示。



图 2-42 应用程序文件夹属性

点击【添加】，出现图 2-43。

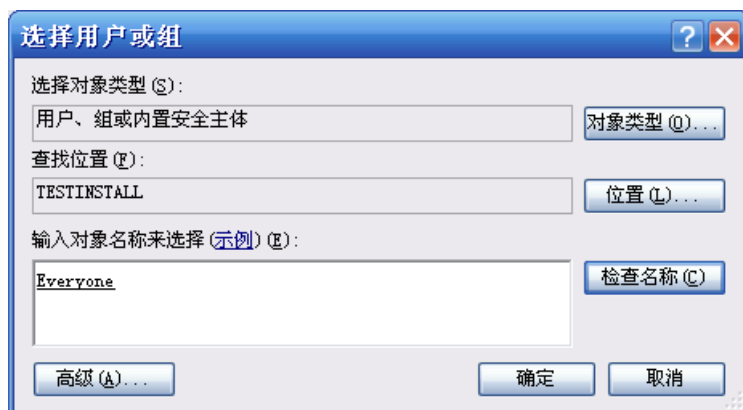


图 2-43 添加用户或组

在“输入对象名称来选择”下输入“everyone”，点击【检查名称】，出现上图信息，点击【确定】，完成用户添加。如图 2-44。

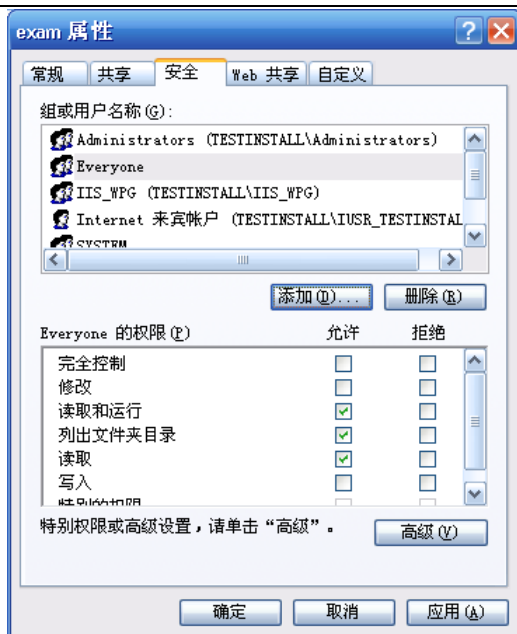


图 2-44 添加用户 everyone

选中 Everyone 用户，将其权限选为“完全控制”，点击【确定】，完成 exam 文件夹权限修改。

至此，Windows Server 2003 系统下的 IIS 服务配置全部完成，过程中可能出现各种各样的问题，希望学者能够仔细分析，并找出解决办法。

#### 小提示：

很多同学平时使用的都是 Windows XP，会问能不能在 XP 系统上部署 IIS 服务，回答是肯定的。Windows XP 同样提供了 IIS 服务功能，该服务默认是关闭，需要用户自行安装，这里不做讲解。具体方法可参考 Windows Server 2003 的 IIS 安装配置方法，但我们不建议使用 XP 作为 IIS 的服务器，因为 XP 系统本身有十个连接数的限制。一个 web 应用程序，一般不止 10 个用户，所以 XP 系统部署 IIS 服务器不适合。



## 本章练习

- 1、学习操作系统的安装，并掌握虚拟机的使用方法。
- 2、从网络上找一个 ASP 应用系统，学习 IIS 服务器的配置。

## 第 3 章 BUG 管理及测试管理工具

### 本章要点

本章主要讲述软件测试工作的基石：**Bug** 管理流程及管理工具。结合前面两章的知识，从理论到实际逐步将知识融会贯通。读者在本章将学会如何利用测试工具进行软件测试工作，加深对测试理论的理解，并掌握软件测试工作所需的基本技术能力。

### 3.1 Bug 管理流程及工具介绍

在软件测试工作过程中，始终围绕一个主题是软件的 **Bug** 管理流程。软件测试人员的职责是找出软件系统中存在的各种缺陷，以及跟踪处理这些缺陷。所以，从测试管理角度来讲，如何有效的管理我们发现的 **Bug** 将直接影响软件的质量与我们的工作效率。

什么是 **Bug** 我们在第一章中已经介绍过，这里就不再重复。大多数公司采用下面的 **Bug** 处理方式，这样的方式从测试管理角度来讲，起到了层层监控、步步把关的作用。下面详细分析一下该流程。

**Bug** 管理流程是以参与人员的角色分工的，我们这里主要从测试人员、测试组长、开发组长、开发人员、项目经理等角色考虑。

#### 测试人员

测试人员发现 **Bug** 后，利用 **Bug** 管理工具添加 **Bug**，此时 **Bug** 的状态为 **new**（新建），在添加 **Bug** 时需指定该 **Bug** 的下步处理人，一般情况下为当前项目的测试组长。

#### 测试组长

测试组长查看对应项目中需要自己处理的 **Bug**，进行 **Bug** 的 **Review**（审查）工作，检查测试组员新增的 **Bug** 是否符合规范，比如语言描述是否清晰；问题定位是否正确等，或者判断该问题是否确实是一个 **Bug**，还是因组员不熟悉需求、理解偏差而引起的误提。如有问题，则将该 **Bug** 返回给 **Bug** 提交者，让其修改后再提交给测试组长。如无问题，则将该 **Bug** 提交给开发组长。

## 开发组长

开发组长处理指派给其的 Bug。根据 Bug 的所属功能模块标识，分派给相应的开发人员，如果在这个过程中，开发组长认为某些 Bug 提交的有问题，则可返回给测试组长，并加上相应的说明（Comment），由测试组长再次审查。测试组长则会与测试组员共同处理该 Bug。如无问题，则可再次指派给开发组长。

## 开发人员

开发人员处理开发组长指派给其的 Bug。根据 Bug 的描述重现并修复 Bug，修复后将该 Bug 状态置为“Fixed”（已修复），并指派给相应的 Bug 提交者，由 Bug 提交者在后续版本中进行验证。如果开发人员认为该问题不是一个 Bug，则可向开发组长反映，或者咨询 Bug 的提交者；测试人员将开发人员 Fix 回来的 Bug 进行验证，如果该 Bug 被成功修复，则关闭该 Bug，如果经检查未能成功修复，则 Reopen（重新打开）该 Bug，继续按照 Bug 的处理流程流转。

## 项目经理

当对提交的 Bug 有分歧时，可由项目经理、测试组长、开发组长等进行 Bug 的评审，并商定问题如何处理，是否作为保留或是当前版本不改，需给出一个定论。

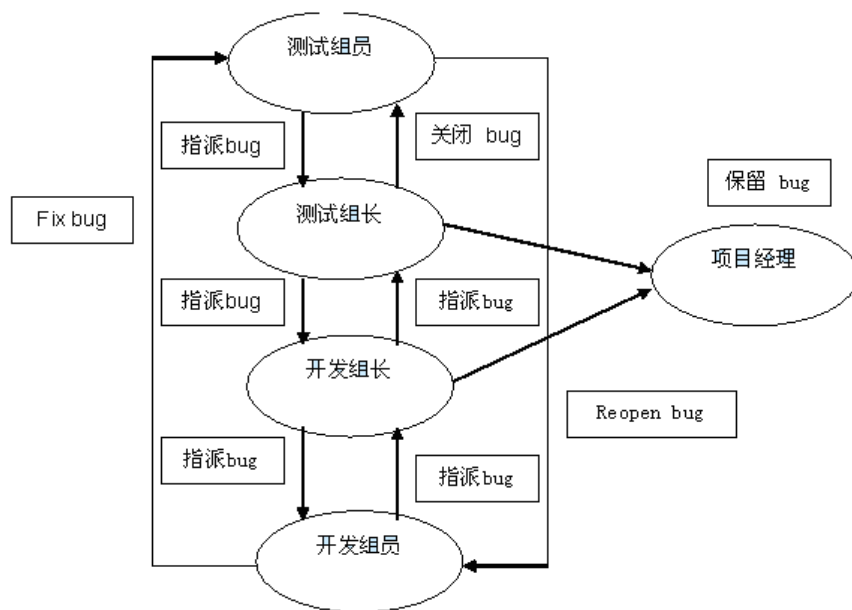


图 3-1 Bug 处理流程

一般来讲，Bug 的处理是一个循环反复的过程。当出现争议的时候，必须由项目负责人参与 Bug 的处理，而不能由开发组或者测试组单方面决定 Bug 的终止。

注意：

任何人都不应具备删除 Bug 的权限。

有了清晰的 Bug 管理流程，如果再辅以先进完善的 Bug 管理工具，软件测试的工作就能顺利地展开，并能在一定程度上提高我们的工作效率。那么，一般有哪些 Bug 管理的工具呢？

目前主要的缺陷管理工具有 TestDirector、Bugzilla、TestTrack、Mantis 等等，有些公司自己开发了一些管理工具。其实不管采用什么工具，只要能高效的协同软件测试工作就行。本教材主要讲两种 Bug 的管理工具，一个是免费开源的 Bugzilla，另外一个收费的，但更专业的测试管理工具 TestDirector。

## 3.2 Bugzilla 工具安装配置

Bugzilla 是一个缺陷跟踪系统，用于对软件产品程序开发过程的缺陷跟踪。它的强大功能表现在以下几个方面：

- 强大的检索功能
- 用户可配置的通过 Email 公布 Bug 变更
- 历史变更记录
- 通过跟踪和描述处理 Bug
- 附件管理
- 完备的产品分类方案和细致的安全策略
- 安全的审核机制
- 强大的后端数据库支持
- Web, Xml, Email 和控制界面
- 友好的网络用户界面
- 丰富多样的配置设定
- 版本间向下兼容

Bugzilla 是一个拥有强大功能的缺陷跟踪系统，它可以使我们更好地在软件开发过程中跟踪处理软件的缺陷，为开发和测试工作以及产品质量的度量提供数据支持，从而有效的保证软件产品的质量。针对很多用户无法成功安装配置 Bugzilla 的现象，本章重点讲解如何在 Windows 2003 Server 上安装配置 Bugzilla 工具及搭建一个简单 SMTP 服务器。Bugzilla 的使用方法比较简单，我们这里将不再介绍，大家可以网络上找到相关的资料自行学习，亦可参考其说明文档，学习 Bugzilla 的使用方法。

### 3.2.1 Bugzilla 安装配置

Bugzilla 一般在 Linux 平台上安装使用，当然也可 Windows 平台上安装，它的

安装过程较为复杂，需要 ActivePerl、Mysql 软件支持，以 IIS 或者 Apache 作为 web 服务器，如果公司无邮件服务器，还需配置一个局域网内的邮件服务器。下面我们来介绍一下 Bugzilla 在 Windows 2003 Server 上的安装配置。在安装开始之前，建议大家在平时的学习、测试活动中，设置密码时不要将密码设置的太复杂，避免遗忘密码，并且也没有必要。

约定：

本次安装所需软件版本分别为：

ActivePerl: ActivePerl -5.8.8.822-MSWin32-x86-280952.msi

Mysql: MySql-5.0.18.exe

Bugzilla: Bugzilla-3.1.3.tar.gz

Bugzilla 中文包: Bugzilla-3.1.2-cn.zip

### 安装 ActivePerl

1、双击 ActivePerl -5.8.8.822-MSWin32-x86-280952.msi，点 图 3- 2 中的【Next】。

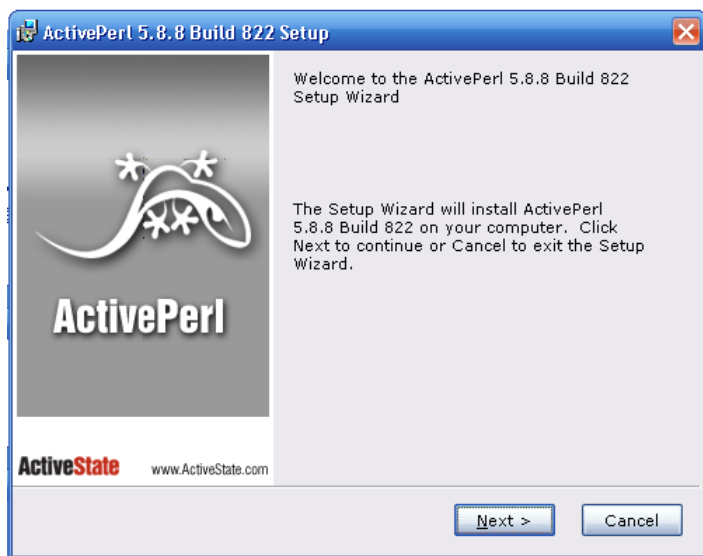


图 3- 2 ActivePerl 准备安装界面

2、选择 “I accept...” ， 点击 【Next】 。

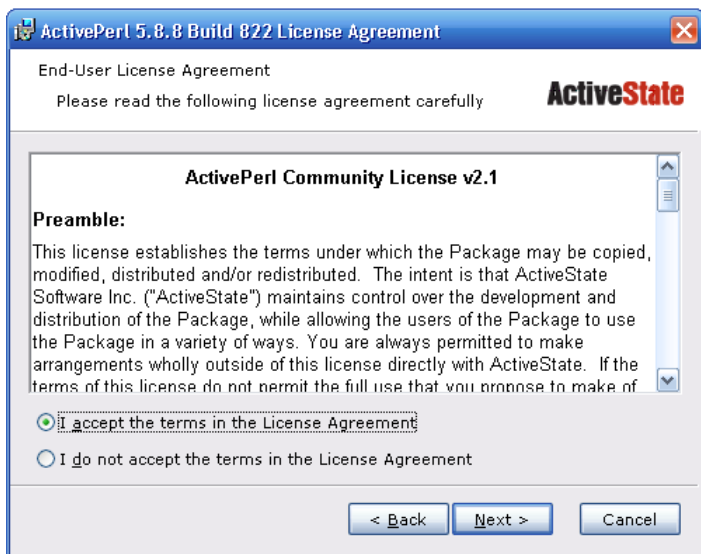


图 3-3 ActivePerl 许可证界面

3、默认设置，点击【Next】。

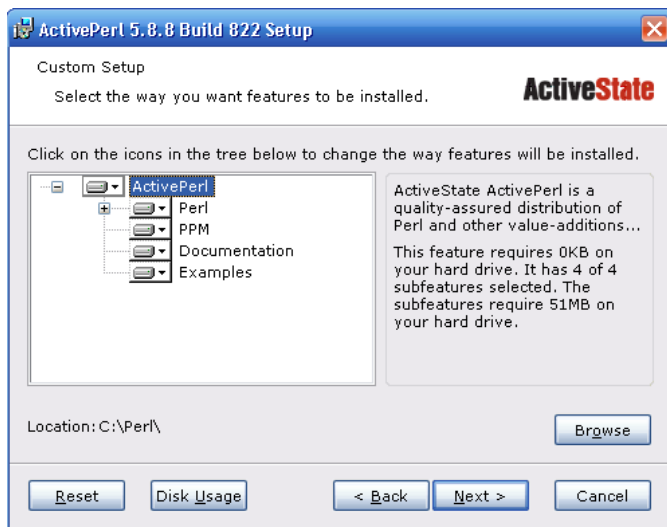


图 3-4 设置安装目录界面

4、默认设置，点击【Next】。

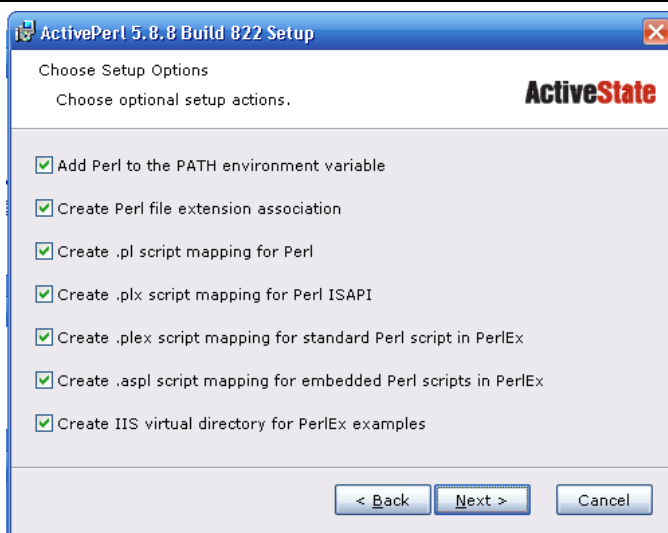


图 3- 5 安装选项设置

5、点击【Install】，开始安装。

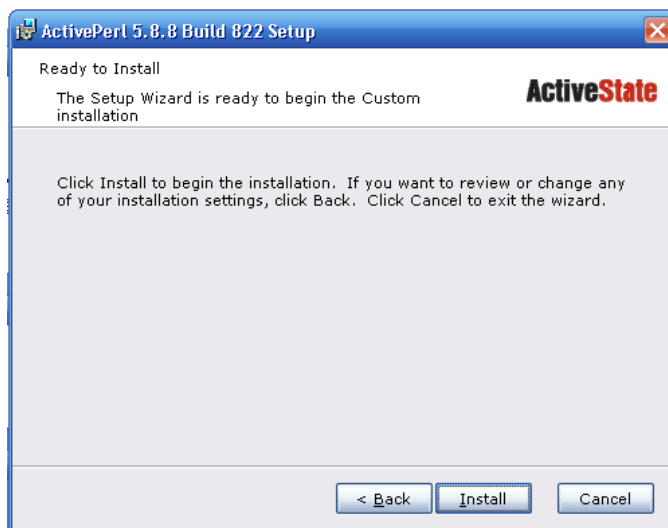


图 3- 6 准备安装界面

6、安装过程。

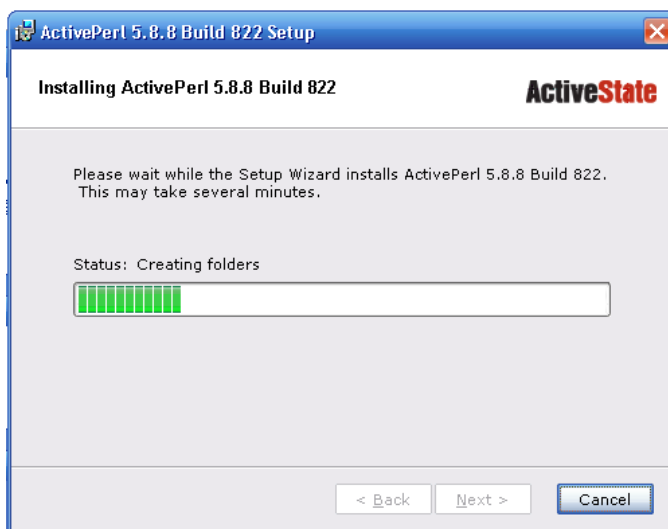


图 3-7 安装进程

7、取消“Display the...”前的勾，点击【Finish】，完成 ActivePerl 的安装。



图 3-8 完成安装

### 安装配置 Mysql

1、双击 MySQL-5.0.18.exe，点击 图 3-9 中的【Next】。



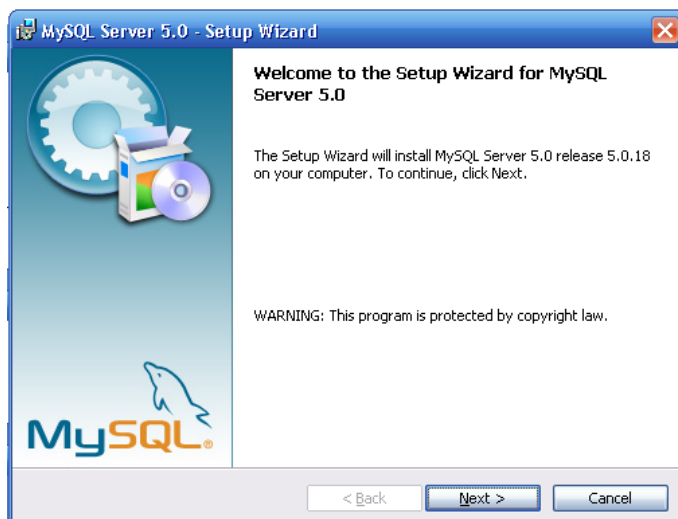


图 3- 9 MySql 准备安装

2、选择“Custom”，点击【Next】。

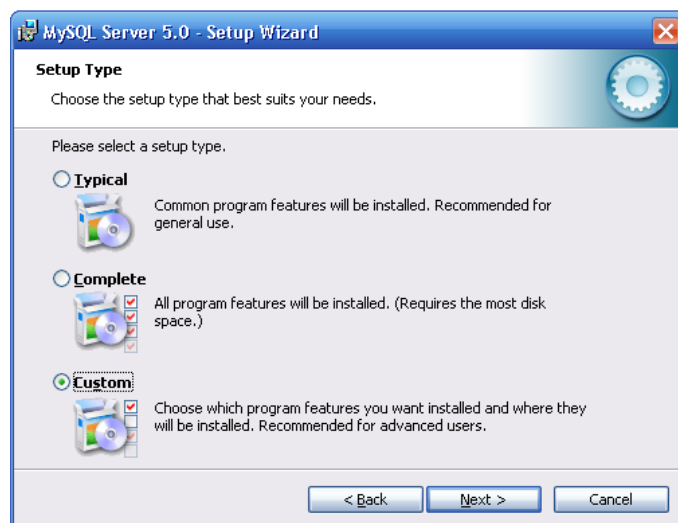


图 3- 10 安装类型选择

3、点击【Change...】，修改安装路径。

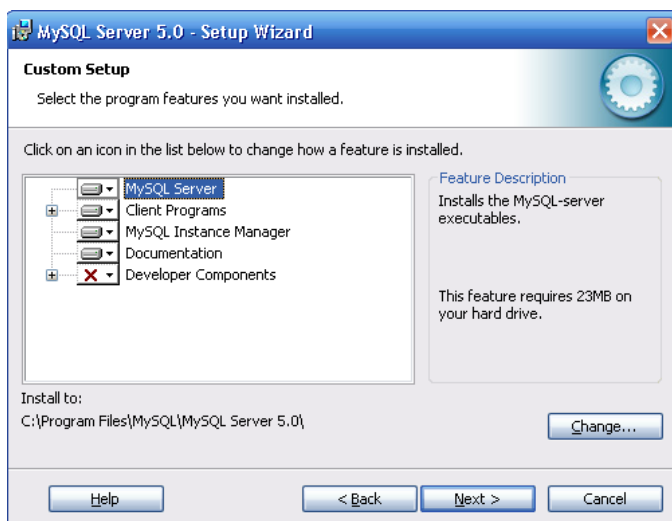


图 3- 11 安装路径设置

4、按图 3- 12 所示，修改安装路径，建议修改为某系统盘的根目录，便于管理，比如此处设置为“C:\mysql”，设置完成后点击【OK】。

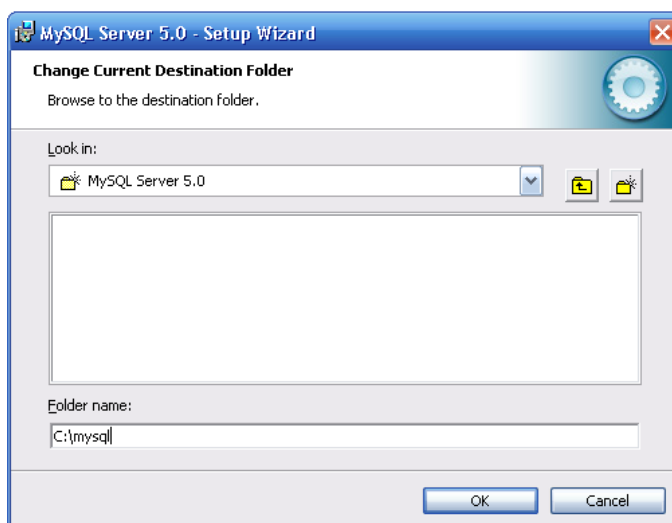


图 3- 12 修改安装路径

5、修改安装路径后，点击【Next】。

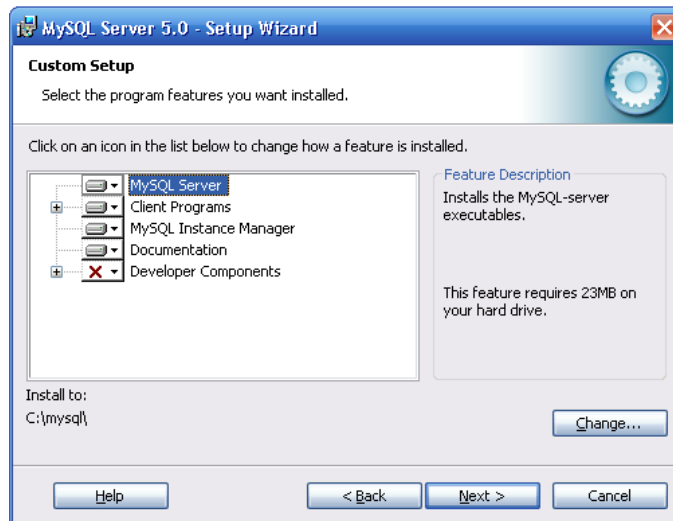


图 3- 13 继续安装

6、点击【Install】，执行安装。

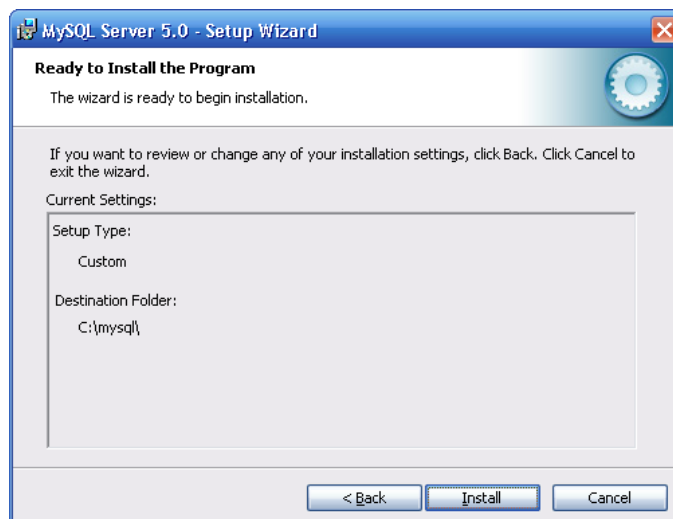


图 3- 14 开始安装

7、安装进行中。

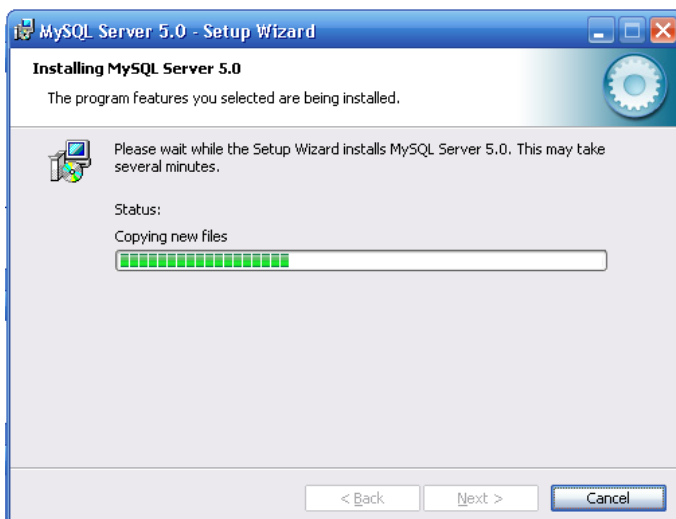


图 3-15 安装进程

8、选择“Skip Sign-Up”，跳过注册，点击【Next】。



图 3-16 MySql 注册界面

9、选中勾，点击【Finish】，进入 Mysql 配置界面。

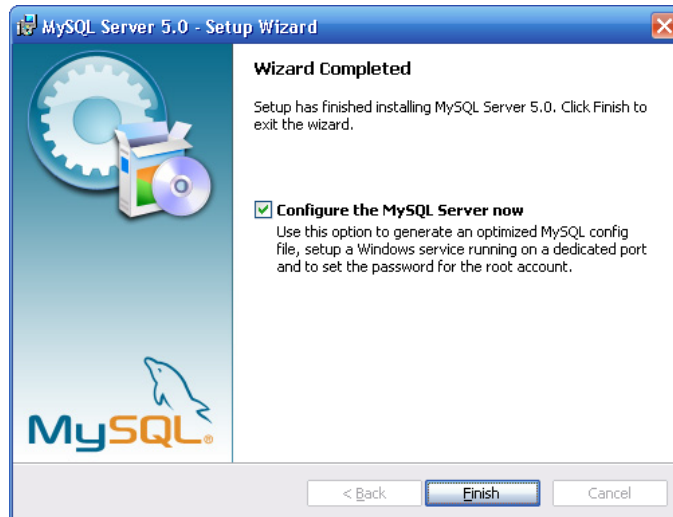


图 3- 17 完成安装

10、该界面是 Mysql 配置欢迎界面，点击【Next】。

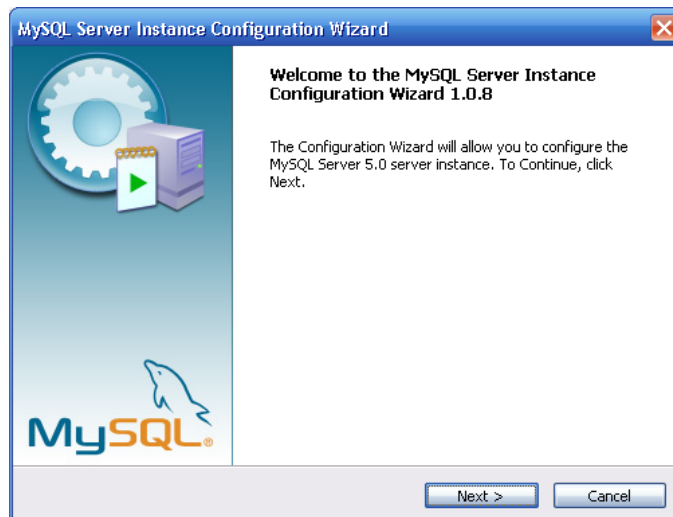


图 3- 18 进入配置界面

11、按 图 3- 19 所示，选择 “Standard...”（标准配置），点击【Next】。

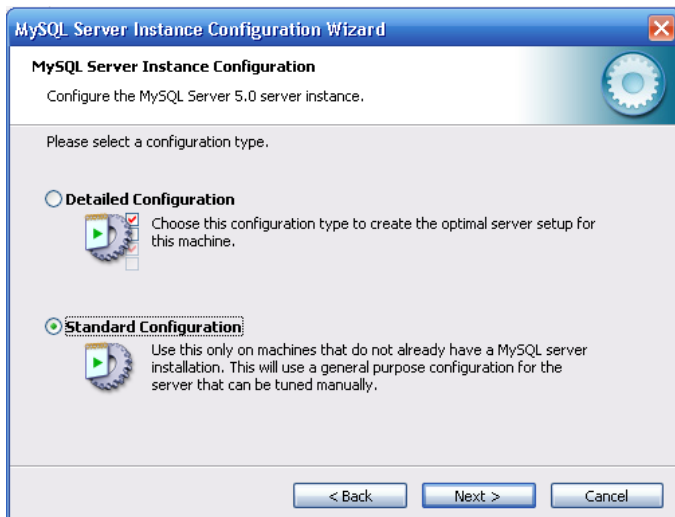


图 3- 19 选择配置类型

12、按 图 3- 20 所示设置，点击【Next】。

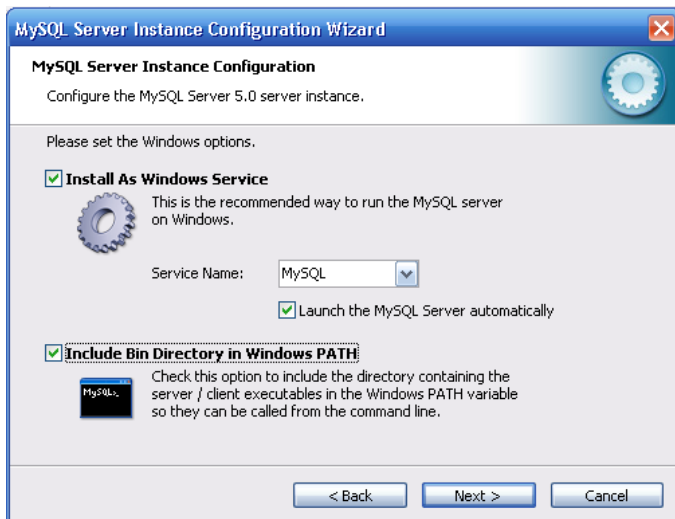


图 3- 20 配置 MySql 界面

13、勾选所有项，并设置 root 用户密码，点击【Next】。



图 3-21 设置 MySql 帐号信息

14、点击【Execute】，执行配置。

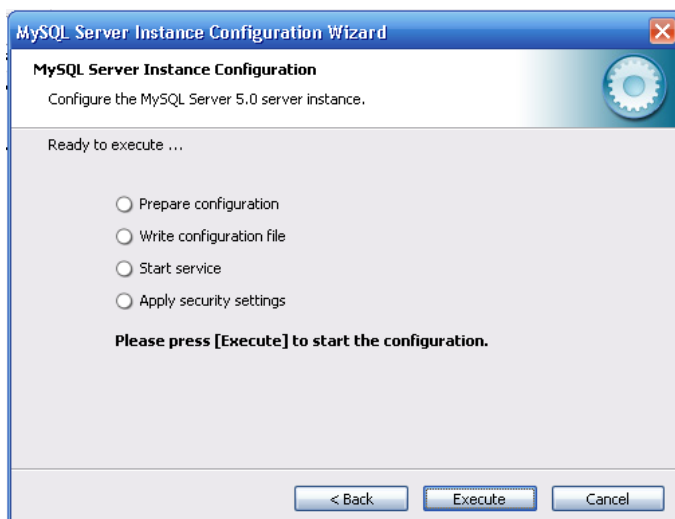


图 3-22 MySql 配置界面

15、点击【Finish】完成 Mysql 的安装与配置。

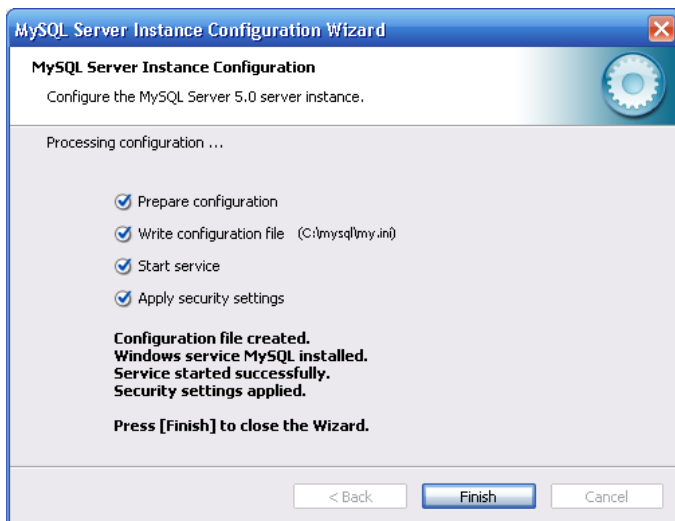


图 3-23 MySQL 安装完成

16、打开命令提示符，进入mysql的bin目录，如图 3-24 所示。

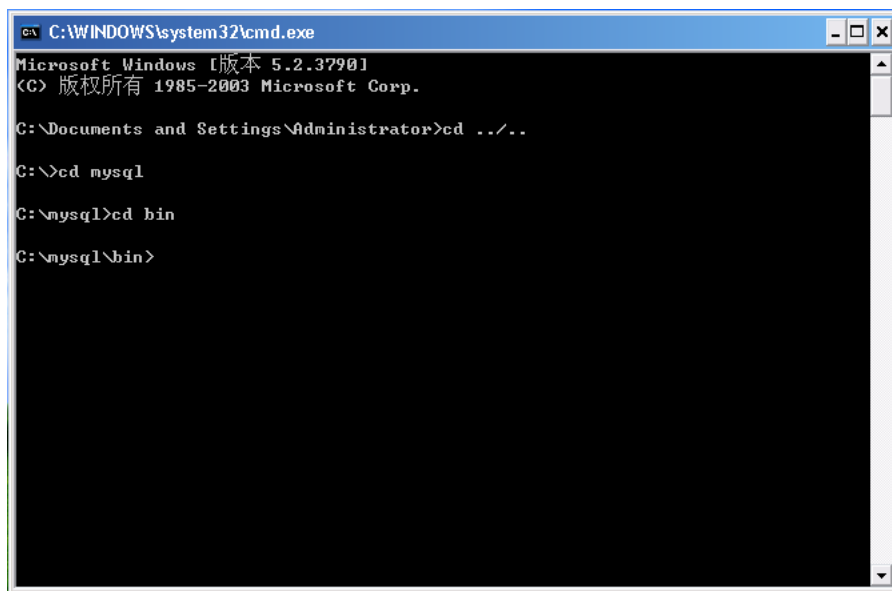


图 3-24 进入 MySQL bin 目录

17、登录Mysql，输入mysql -u root -p，此命令意思是以root登录，并要求输入密码，输入密码后，界面如图 3-25 所示。



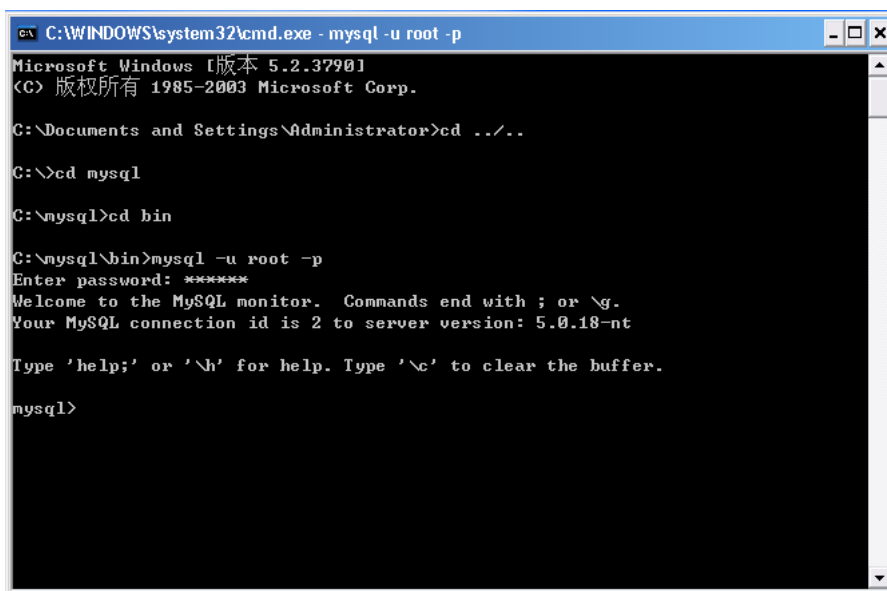


图 3-25 登录 MySQL

18、成功登录到 mysql 后，创建 bugs 库，并创建使用该库的用户 bugs，密码为 123456。具体命令方法：

- a、建立数据库 bugs：create database bugs;
- b、建立用户 bugs，密码为空，对 Bugs 数据库授予一定权限：  
GRANT  
SELECT,INSERT,UPDATE,DELETE,INDEX,ALTER,CREATE, LOCK  
TABLES,DROP,REFERENCES ON bugs.\* TO bugs@localhost  
IDENTIFIED BY '123456';  
其中“bugs@localhost”中的“bugs”为用户名，“IDENTIFIED BY  
'123456'”中的“‘123456’”为 bugs 用户的密码。
- c、从 mysql 数据库授权表中重新装载权限：FLUSH PRIVILEGES;

操作过程如图 3-26 所示。

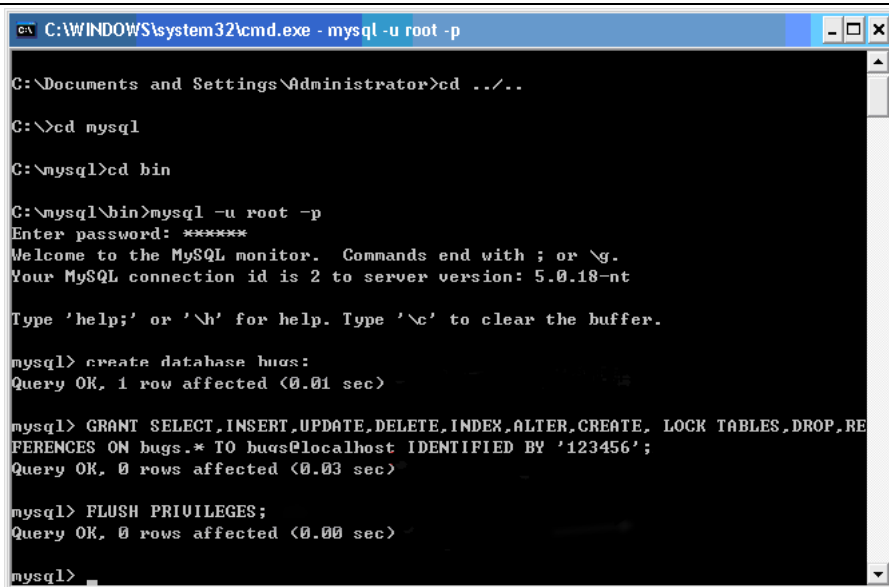


图 3-26 创建 bugs 数据及用户

### 解压部署 Bugzilla

将bugzilla-3.1.3.tar.gz中的bugzilla-3.1.3 解压放到C盘根目录下，并改名为bugzilla，完成目录路径为C:\bugzilla，如图 3-27 所示。

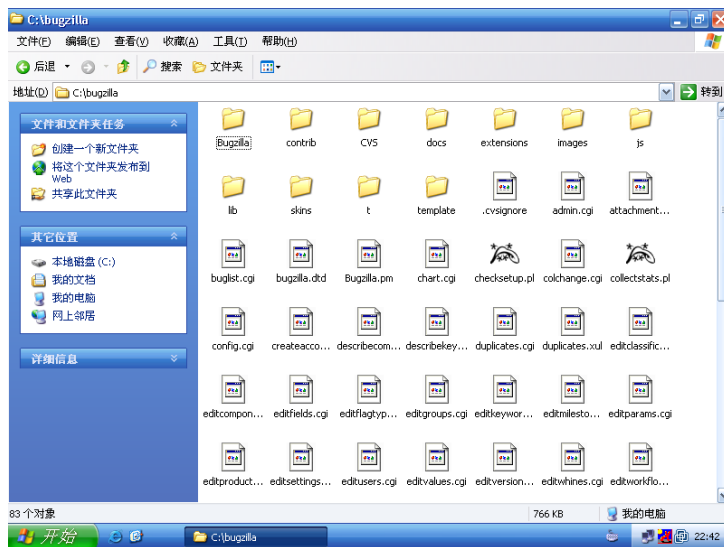


图 3-27 bugzilla 目录

### 配置 IIS 服务

1、打开控制面板->管理工具-> Internet 信息服务(IIS)管理器，选择“默认网

站”，右键选择“属性”，打开“主目录”，点击【配置】，在“应用程序扩展”中点击【添加】，在“可执行文件”中输入 `C:\Perl\bin\perl.exe -T "%s" %s`，在“扩展名”中输入：`.cgi` 在“限制为”中输入：`GET,HEAD,POST`，点击【确定】，完成 `cgi` 扩展添加。

在“应用程序扩展”中修改 `.pl` 的可执行文件为“`C:\Perl\bin\perl.exe -T "%s" %s`”，原来的是“`C:\Perl\bin\perl.exe "%s" %s`”，其他不变。选择“默认网站”，右键选择“新建”下的“虚拟目录”，出现图 3-28，点击【下一步】。

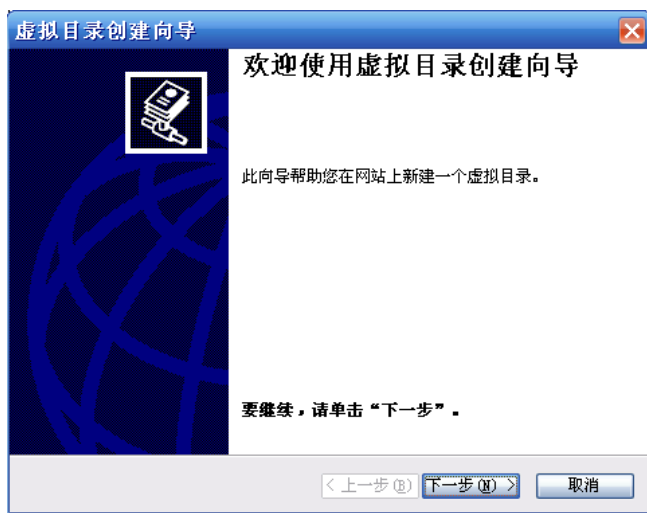


图 3-28 创建虚拟目录

2、输入别名“bugzilla”，点击【下一步】。

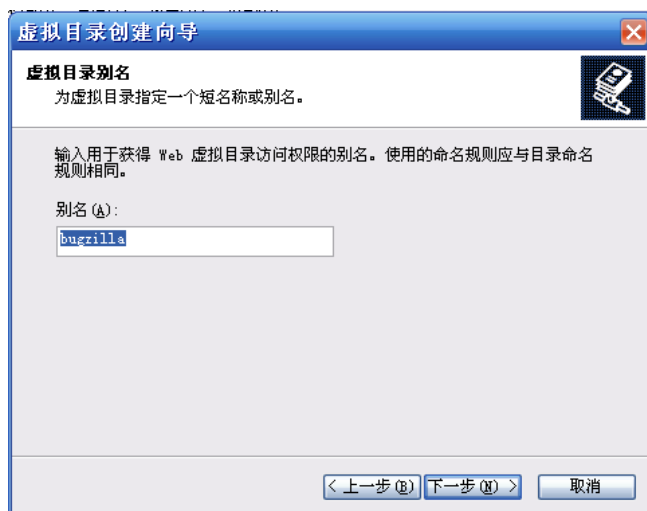


图 3-29 输入虚拟目录名称

3、设置虚拟目录路径，这里选择 bugzilla 的目录，点击【下一步】。

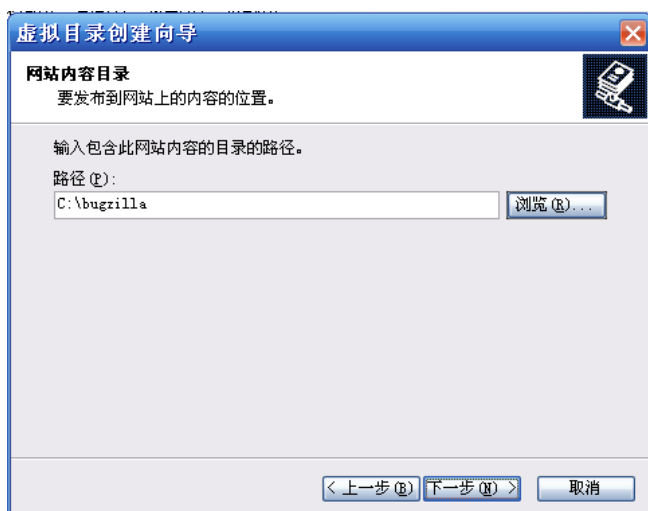


图 3-30 设置虚拟机目录路径

4、按图 3-31 设置，点击【下一步】，完成虚拟目录设置。

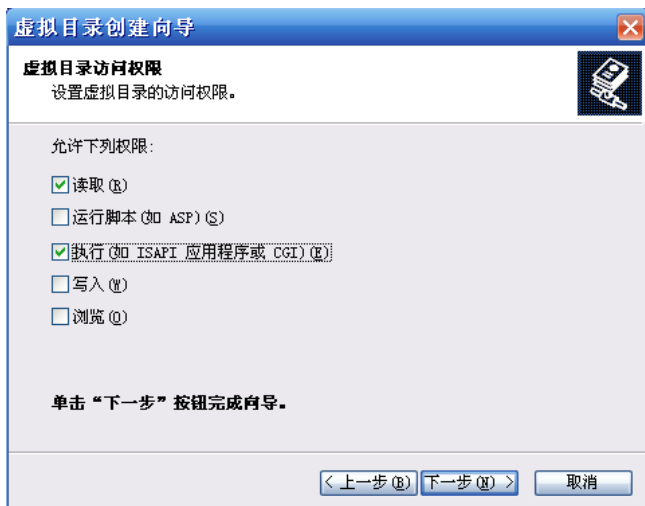


图 3-31 设置虚拟目录执行权限

5、选择刚才建立的虚拟目录bugzilla,右键选择“属性”下的“文档”。在默认内容文档中增加index.cgi。如图 3-32 所示。

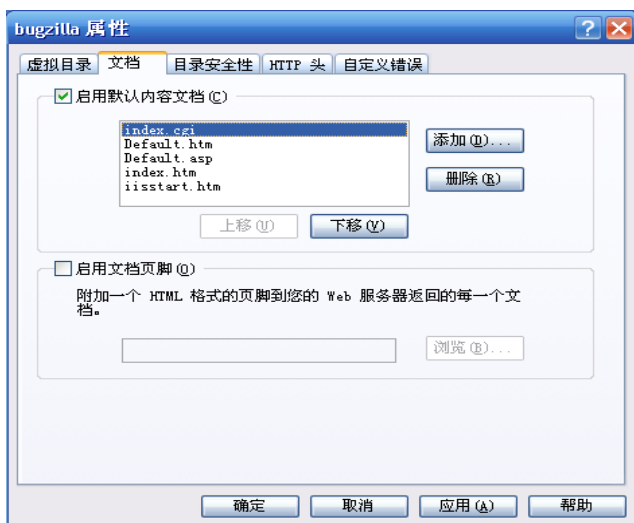


图 3-32 设置虚拟目录文档属性

6、点击“Web服务扩展”，启用相关扩展服务，如图 3-33。

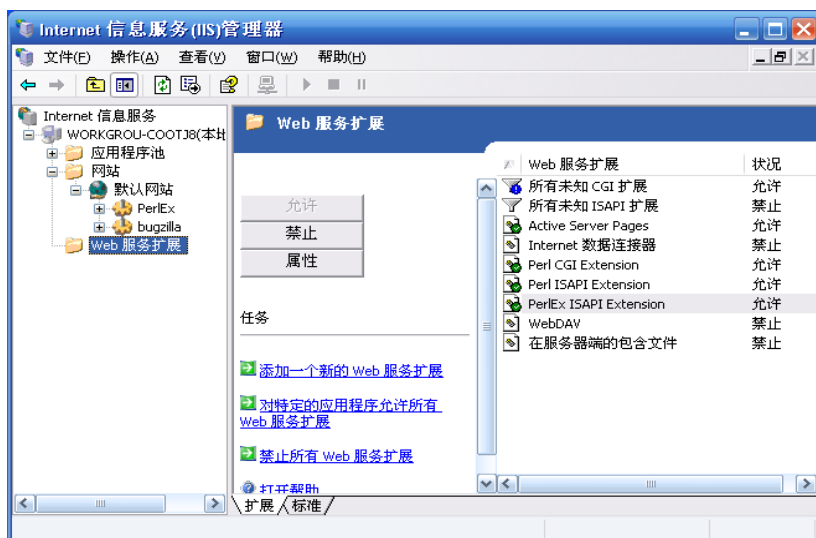


图 3-33 Web 服务扩展

## 搭建邮件服务器

### 1、安装 POP3 服务组件

以系统管理员身份（administrator）登录 Windows Server 2003 系统。依次进入“控制面板→添加或删除程序→添加/删除 Windows 组件”，在弹出的“Windows 组件向导”对话框中选中“电子邮件服务”选项，点击【详细信息】按钮，可以看到该选项包括两部分内容：POP3 服务和 POP3 服务 Web 管理。为方便用户远程 Web 方式管理邮件服务器，建议选中“POP3 服务 Web 管理”。

## 2、安装 SMTP 服务组件

选中“应用程序服务器”选项，点击【详细信息】按钮，接着在“Internet 信息服务(IIS)”选项中查看详细信息，选中“SMTP Service”选项，最后点击【确定】按钮。此外，如果用户需要对邮件服务器进行远程 Web 管理，一定要选中“万维网服务”中的“远程管理(HTML)”组件。完成以上设置后，点击【下一步】按钮，系统就开始安装配置 POP3 和 SMTP 服务了。

## 3、配置 POP3 服务器

### 创建邮件域

点击“开始→管理工具→POP3 服务”，弹出 POP3 服务控制台窗口。选中左栏中的 POP3 服务后，点击右栏中的“新域”，弹出“添加域”对话框，接着在“域名”栏中输入邮件服务器的域名，也就是邮件地址“@”后面的部分，如“xkh.com”，最后点击【确定】按钮。如图 3-34 所示。

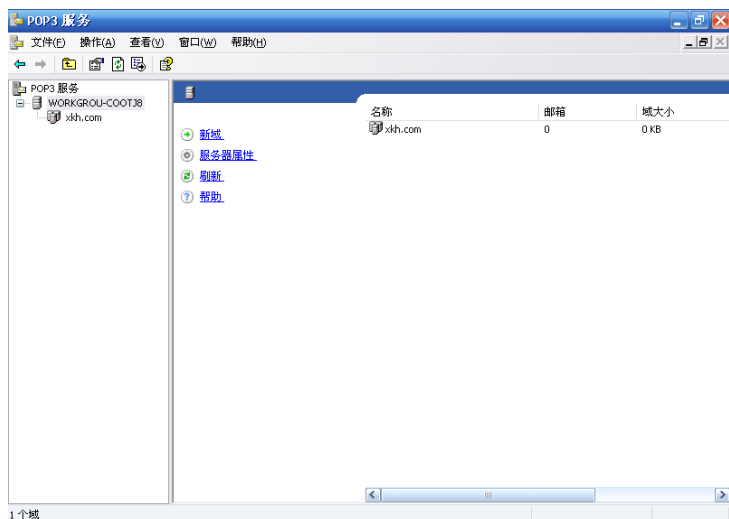


图 3-34 创建邮件域

### 创建用户邮箱

选中刚才新建的“xkh.com”域，在右栏中点击“添加邮箱”，弹出添加邮箱对话框，在“邮箱名”栏中输入邮件用户名，然后设置用户密码，最后点击【确定】按钮，完成邮箱的创建。如图 3-35 所示。

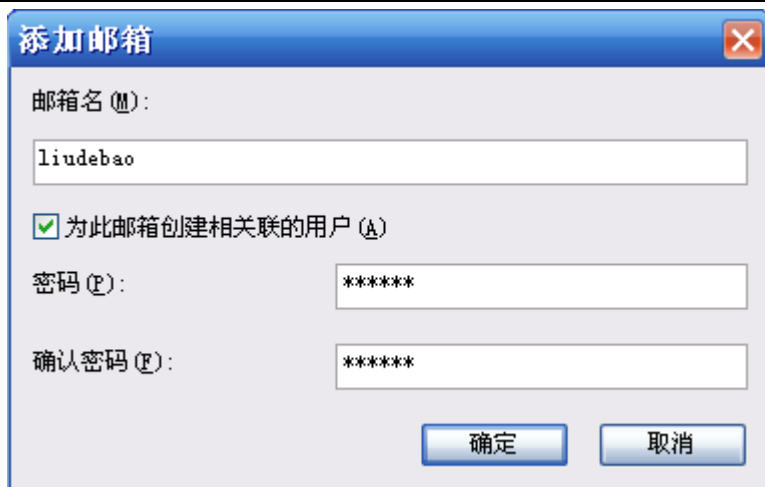


图 3-35 创建邮箱用户

成功界面如图 3-36 所示。

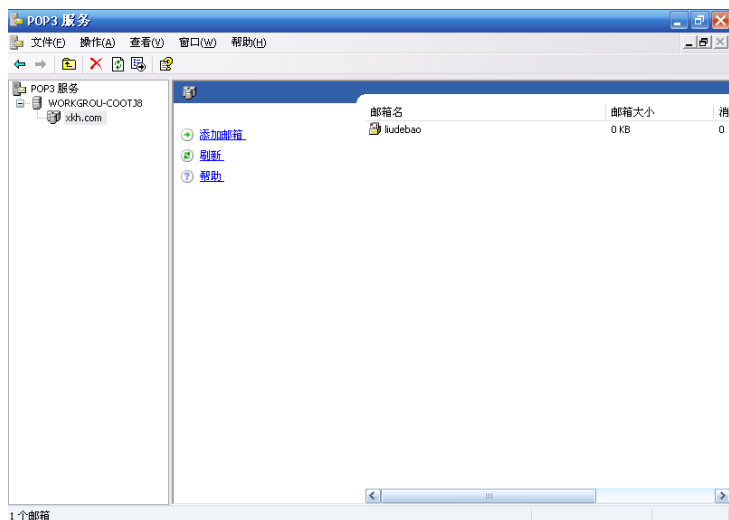


图 3-36 用户创建成功

#### 4、配置 SMTP 服务器

完成 POP3 服务器的配置后,就可开始配置 SMTP 服务器了。点击“开始→程序→管理工具→Internet 信息服务(IIS)管理器”,在“IIS 管理器”窗口中右键点击“默认 SMTP 虚拟服务器”选项,在弹出的菜单中选中“属性”,进入“默认 SMTP 虚拟服务器”窗口,切换到“常规”标签页,在“IP 地址”下拉列表框中选中邮件服务器的 IP 地址即可。点击【确定】按钮,这样一个简单的邮件服务器就架设完成了。

完成以上设置后,用户就可以使用邮件客户端软件连接邮件服务器进行邮件收

发工作了。在设置邮件客户端软件的 SMTP 和 POP3 服务器地址时，输入邮件服务器的 IP 即可。

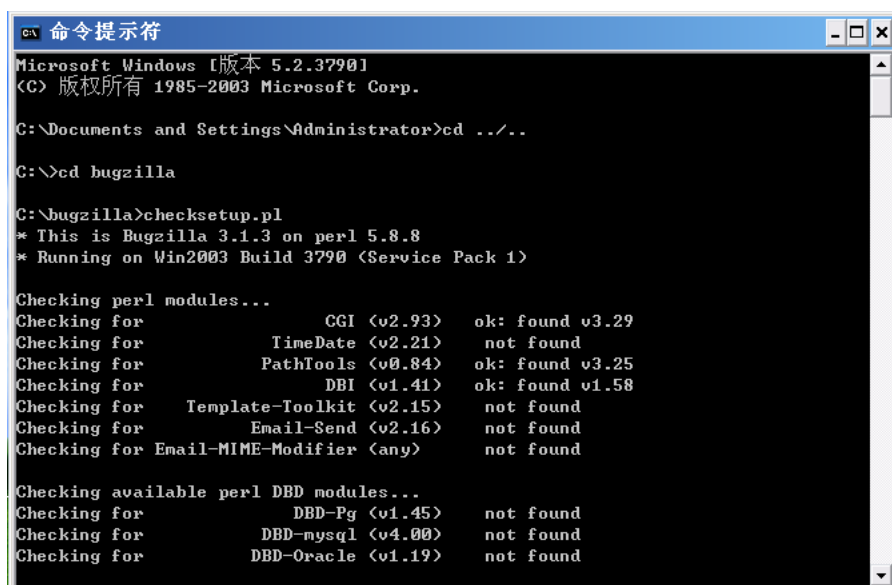
### 5、远程 Web 管理

Windows Server 2003 还支持对邮件服务器的远程 Web 管理。在远端客户机中，运行 IE 浏览器，在地址栏中输入“https://服务器 IP 地址: 8098”，将会弹出连接对话框，输入管理员用户名和密码，点击“确定”按钮，即可登录 Web 管理界面。

### 安装 perl 模块

上面步骤都正确无误后，我们开始 bugzilla 所需 perl 模块的安装。

1、打开命令提示符，进入 bugzilla 目录，然后运行 checksetup.pl 检查所缺模块，如图 3-37 所示。



```

命令提示符
Microsoft Windows [版本 5.2.3790]
(C) 版权所有 1985-2003 Microsoft Corp.

C:\Documents and Settings\Administrator>cd ../../

C:\>cd bugzilla

C:\bugzilla>checksetup.pl
* This is Bugzilla 3.1.3 on perl 5.8.8
* Running on Win2003 Build 3790 (Service Pack 1)

Checking perl modules...
Checking for      CGI <v2.93>      ok: found v3.29
Checking for      TimeDate <v2.21> not found
Checking for      PathTools <v0.84> ok: found v3.25
Checking for      DBI <v1.41>      ok: found v1.58
Checking for      Template-Toolkit <v2.15> not found
Checking for      Email-Send <v2.16> not found
Checking for      Email-MIME-Modifier <any> not found

Checking available perl DBD modules...
Checking for      DBD-Pg <v1.45>    not found
Checking for      DBD-mysql <v4.00> not found
Checking for      DBD-Oracle <v1.19> not found
  
```

图 3-37 检查 perl 缺失模块

2、如果缺少所需模块，会显示“no found”字样，安装时，需将缺少的模块都安装上。在上面信息中，bugzilla 已经提示了缺少哪些文件，所有以“ppm”开头的都为缺少项，必须都安装，比如“ppm install TimeDate”。右键点击命令提示符窗口，选择“标志”，然后选中需拷贝的信息，如“ppm install TimeDate”，再点击右键，使用“粘贴”即完成命令输入，回车后进行安装。如图 3-38 所示。



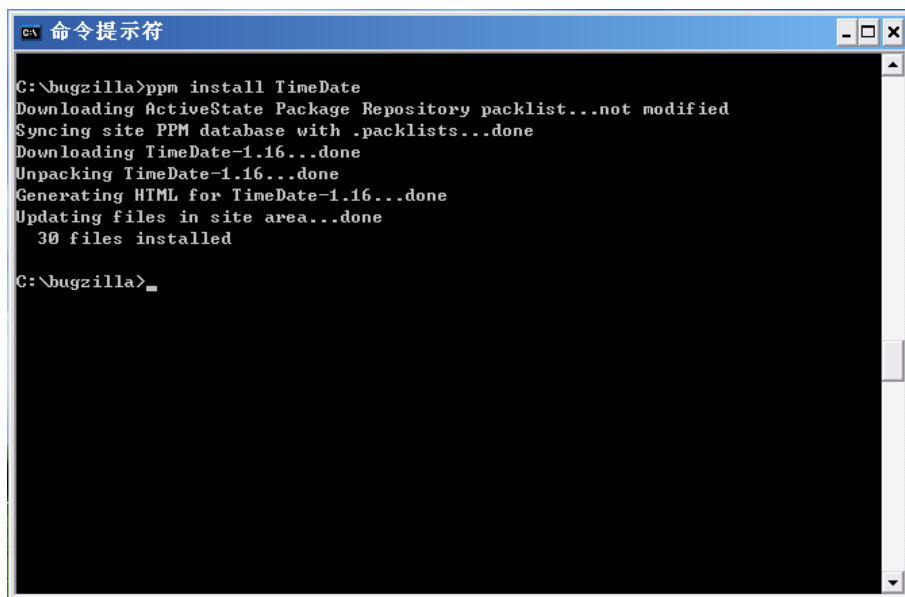


图 3- 38 安装缺失 perl 模块

### 修改 localconfig 文件

所有模块安装完成后，再次运行 checksetup.pl，检查是否还有遗漏的模块未安装，如果没有，则提示要修改 localconfig 文件：

This version of Bugzilla contains some variables that you may want to change and adapt to your local settings. Please edit the file `./localconfig` and rerun `checksetup.pl`.

The following variables are new to `./localconfig` since you last ran `checksetup.pl`: `create_htaccess`, `webservergroup`, `db_driver`, `db_host`, `db_name`, `db_user`, `db_pass`, `db_port`, `db_sock`, `db_check`, `index_html`, `cvsbin`, `interdiffbin`, `diffpath`

进入 `C:\bugzilla` 中打开 `localconfig`，修改其中的数据库密码：

```
# The name of the database
$db_name = 'bugs';

# Who we connect to the database as.
$db_user = 'bugs';

# Enter your database password here. It's normally advisable to specify
# a password for your bugzilla database user.
# If you use apostrophe (') or a backslash (\) in your password, you'll
```

```
# need to escape it by preceding it with a '\' character. (\') or (\)
# (Far simpler just not to use those characters.)
$db_pass = ";---此处修改为实际 bugs 用户的密码，如上面设置的
“123456”。
```

修改完成后保存，再次在命令提示符中运行 checksetup.pl，如无问题，则开始相关信息的配置，如下列信息所示：

设置 SMTP 服务器，这里填上我们前面配置的邮件服务器地址：

```
Bugzilla requires an SMTP server to function on Windows.
Please enter your SMTP server's hostname: 192.168.0.102
```

设置一个超级管理员邮箱地址，即为 bugzilla 设置一个超级管理员：

```
Precompiling templates...done.

.....

Looks like we don't have an administrator set up yet.
Either this is your first time using Bugzilla, or your
administrator's privileges might have accidentally been deleted.

Enter the e-mail address of the administrator: liudebao@xkh.com
```

输入该超级管理员的真实姓名：

```
Enter the real name of the administrator: liudebao
```

设置超级管理员密码：

```
Enter a password for the administrator account: 123456
```

再次输入超级管理员密码：

```
Please retype the password to verify: 123456
```

其他配置信息：

```
liudebao@xkh.com is now set up as an administrator.
Creating default classification 'Unclassified'...
Creating initial dummy product 'TestProduct'...

Now that you have installed Bugzilla, you should visit the
'Parameters' page (linked in the footer of the Administrator
account) to ensure it is set up as you wish - this includes
setting the 'urlbase' option to the correct URL.
```

完成后，即可在IE中输入<http://localhost/bugzilla>访问，如果配置无误，可见到

图 3- 39 所示的成功界面。

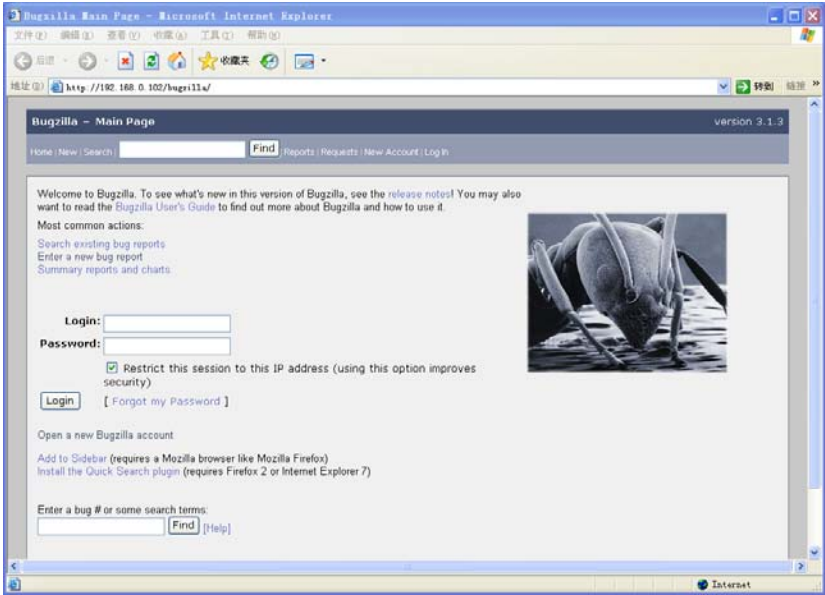


图 3- 39 bugzilla 系统界面

如果没有成功，请检查根据错误提示，检查上述步骤。接下来，我们可以做汉化的操作，汉化方法比较简单，只需将中文包拷贝到Bugzilla所在目录的template下，然后刷新Bugzilla页面即可完成汉化，汉化后的效果如 图 3- 40 所示。



图 3- 40 汉化后的 Bugzilla

### 3.2.2 Bugzilla 使用

Bugzilla的使用方法比较简单，点击图 3-40 界面上的“Bugzilla 用户手册”即可打开Bugzilla的使用说明书，这里不再赘述。

## 3.3 测试件管理

在日常软件测试工作过程中，如何管理测试过程中生产的各种文档资料，提高我们的工作效率，是个非常重要的课题。

一般的测试工作流程是先进进行需求分析，提取出测试需求，然后根据我们提取出的测试需求进行测试功能点的划分，测试用例的编写。在实施测试的时候，利用前期设计好的测试用例进行测试，测试过程中如果有问题，需要提交相应的 Bug。如果过程中还有变动的话，可进行测试需求以及测试用例的变更，使得测试工作更加有效。那么如何进行这些工件的管理呢？

在没有测试管理工具的时候，我们多是采用文档记录的方式，比如用 word 或者 excel 等，记录下我们的测试需求，然后再编写相关的测试用例，执行测试用例，提交 Bug。需要变更的时候，再找到相关的功能模块进行改变。这样会带来一个问题，当我们的测试项目越来越多，测试需求变动非常频繁，测试用例编写复杂的时候，就会变得难以维护。提取测试需求，编写测试用例，是为了更好的处理我们系统中的 Bug，如果这些工作难以维护，成为我们的工作瓶颈，那么他们的存在也就失去了意义。从软件质量保证角度来讲，已经不适用我们的软件开发测试流程。

这里所讲的测试件管理，从狭义来讲，我们只管理软件测试工作过程，不涉及其他方面的，比如人员管理、任务分配、资源统筹等。

一个有经验的测试管理者会利用较好的工具来进行测试的管理，提高测试部门的工作效率。有些公司使用配置管理的工具进行管理，比如使用微软的 Visual SourceSafe，或者 CVS、SVN 等，也可以使用专业的测试管理工具，比如 TestDirector、ClearCase 等。

在本章中我们重点学习利用 TestDirector 测试管理工具进行测试件的管理。

## 3.4 TestDirector 安装配置及使用

本章节主要分为 TestDirector 安装、TestDirector 后台管理、TestDirector 项目定制、TestDirector 前台功能使用等四个方面。

### 3.4.1 TestDirector 介绍

如今的测试技术不断发展，与之相配套的测试工具也在不断发展。本章节中，我们重点学习 Mercury Interactive 公司（已被 HP 公司收购）推出的一款基于 B/S 结构的测试管理工具 TestDirector。新版的 TestDirector 叫 QC（Quality Center）。

TestDirector 工具系统地控制整个测试过程，并创建整个测试工作流的框架和基础，使整个测试管理过程变得更为简单和有组织，能够帮助我们维护一个测试工程数据库，并且能够覆盖被测应用程序功能性的各个方面。在我们工程项目中的每一个测试点都对应着一个指定的测试需求，并专门提供了一个完善的缺陷跟踪系统，它能够让我们跟踪缺陷从产生到最终解决的全过程。TestDirector 通过与邮件系统相关联，缺陷跟踪的相关信息可以被整个应用开发组、QA、客户支持、负责信息系统的人员所共享。TestDirector 提供了与 WinRunner, LoadRunner, QuickTest Professional 等工具、需求和配置管理工具、建模工具的整合功能。TestDirector 能够与这些测试工具很好的无缝链接，TestDirector 指导我们进行需求定义、测试计划、测试执行和缺陷跟踪，即整个测试过程的各个阶段。通过整合所有的任务到应用程序测试中来确保我们的客户收到更高质量的产品。

这款工具从初期的测试需求提取、设计测试用例、执行测试用例，到最终的缺陷管理，提供了一个完整的测试流程管理。如图 3-41 所示：

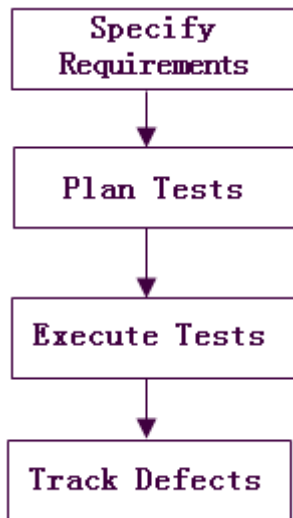


图 3-41 TestDirector 测试管理流程图

#### 测试需求定义（Specify Requirements）

测试需求定义模块分析应用程序并确定测试需求。在测试过程中，我们首先需要弄清楚测试对象，比如主管让你测试什么，有什么要求，有哪些功能点，测试的

程度如何？哪些是关键的业务点等等。弄清楚这些要求，才可以进行针对性的测试需求提取以及测试用例的设计。例如，主管让我们测试《在线考试系统》，按照正规的测试流程，我们需要获取《在线考试系统需求规格说明书》、《在线考试系统概要设计文档》、《在线考试系统详细设计文档》、《在线考试系统测试计划》等等该项目的相关文档，如果没有这样的文档，我们可能需要研发同事提供被测对象的《Function List》（功能列表），从中提取我们的测试需求。

测试需求定义的流程如下 图 3-42：

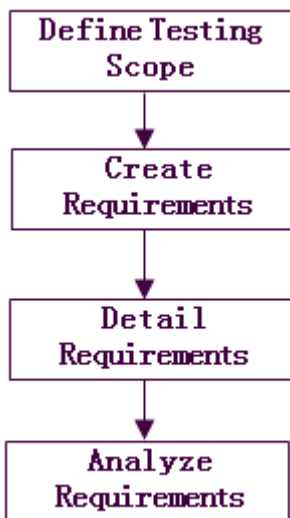


图 3-42 测试需求定义流程图

### 定义测试范围（Define Testing Scope）

检查应用程序文档，并确定测试范围、测试目的、目标和策略。这步我们需要掌握的一个重点是如何确定我们的测试范围。大家知道，测试不是无止境的测试，需要在一定的成本控制范围内开展的。那就需要我们在开展我们测试工作之前先确定一个测试范围，哪些是本次测试的重点，哪些是我们需要关注的。一般来说，系统的关键业务点、用户常用的功能就是我们的测试重点，当然也包括系统的关键业务流程。在这个阶段，我们需要确定好我们的测试范围，参考的文档主要包括需求规格说明书，概要设计文档，详细设计文档，数据库设计文档等等。

### 创建需求（Create Requirements）

创建需求树（Requirements Tree），并确定它涵盖所有的测试需求。这里所谓的需求树，其实就是将被测对象的需求提取出来，以树形结构展示。这样不管是管理还是平时的查看，都很有层次感，便于查找和编辑。

### 描述需求（Detail Requirements）

为“需求树”中的每一个需求主题建立了一个详细的目录，并描述每一个需求，给它分配一个优先级，如有必要的话还可以加上附件。该功能提供了添加附件的功

能，这样我们在实际的工作中就可以将被测需求的相关文档作为附件添加进 TestDirector，从而管理测试过程中所需的工件。

### 分析需求（Analyze Requirements）

利用图表、图形的方式，产生报告和图表来帮助我们分析测试需求，并检查需求以确保它们在你的测试范围内。一般地，我们对于图形方法的报告接受起来比较容易，TestDirector 提供了功能丰富的分析功能，便于我们从多方面管理测试需求。

### 测试计划管理（Plan Tests）

测试计划管理的流程如下 图 3- 43 所示。

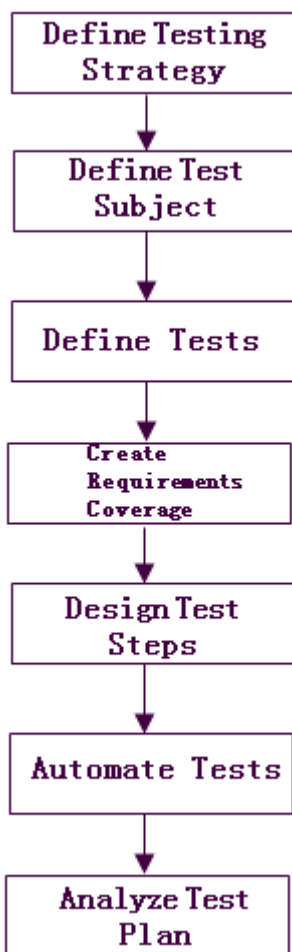


图 3- 43 测试计划管理流程图

### 定义测试策略（Define Testing Strategy）

检查应用程序、系统环境和测试资源，并确认测试目标。软件测试工作的前期，需要检查被测系统的开发进度、文档等，同时要考虑测试资源是否准备妥当，当这

些都没有问题后，我们需要确认一个明确的测试目标，以指导我们的测试工作。

### 定义测试主题（Define Test Subject）

将应用程序基于模块和功能进行划分，并对应到各个测试单元或主题，构建测试计划树（Test Plan Tree）。比如，我们在测试某企业的内部信息管理系统时，需要根据测试的要求，划分出功能性测试、性能测试、安全性测试等等测试主题。这样在测试开展过程中可以做到有的放矢。

### 定义测试（Define Tests）

定义每个模块的测试类型，并为每一个测试创建基本的说明。

### 创建需求覆盖（Create Requirements Coverage）

将每一个测试与测试需求进行连接。

### 设计测试步骤（Design Test Steps）

对于每一个测试，先决定其要进行的测试类型（手动测试和自动测试），若准备进行手动测试，需要为其在测试计划树上创建相应的测试步骤（Test Steps）。测试步骤描述测试的详细操作、检查点和每个测试的预期结果。

### 自动测试（Automate Tests）

对于要进行自动测试的部分，应该利用 MI、自己或第三方的测试工具来创建测试脚本。

### 分析测试计划（Analyze Test Plan）

产生报告和图表来帮助我们分析测试计划数据，并检查所有测试以确保它们满足我们的测试目标。功能设计类似于测试需求定义中的分析功能。

### 测试执行管理（Execute Tests）

测试执行管理流程图如下：

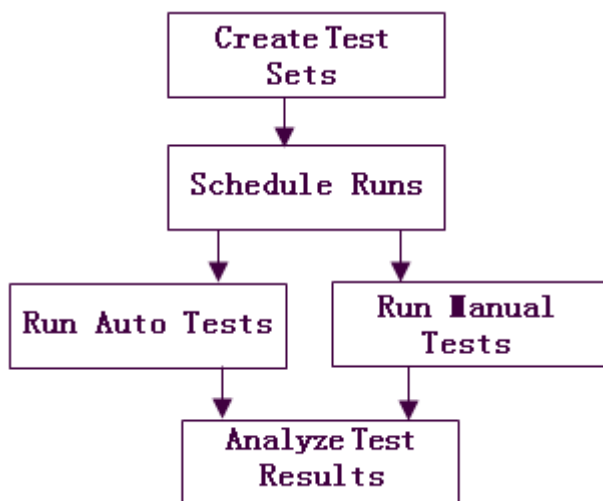


图 3-44 测试执行管理流程图



### 创建测试库（Create Test Sets）

在你的工程中定义不同的测试组来达到各种不同的测试目标，他们可能包括，举个例子，在一个应用程序中测试一个新的应用版本或是一个特殊的功能。并确定每个测试库都包括了哪些测试。

#### 3.4.1.3.2 确定进度表（Schedule Runs）

为测试执行制定时间表，并为测试员分配任务。

#### 3.4.1.3.3 运行测试（Run Tests）

自动或手动执行每一个测试库。

#### 3.4.1.3.4 分析测试结果（Analyze Test Results）

查看测试结果并确保应用程序缺陷已经被发现。生成的报告和图表可以帮助你分析这些结果。

### 缺陷跟踪管理（Tracking Defects）

缺陷跟踪管理流程图如 图 3- 45 所示。

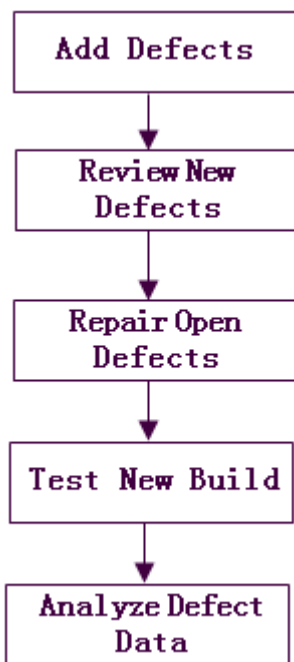


图 3- 45 缺陷跟踪管理流程图

### 创建缺陷（Add Defects）

报告程序测试中发现的新的缺陷。在测试过程中的任何阶段，质量保证人员、开发者、项目经理和最终用户都能创建缺陷。

### 检查新缺陷（Review New Defects）

检查新的缺陷，并确定哪些缺陷应该被修复。

### 修复打开的缺陷（Repair Open Defects）

修复那些我们决定要修复的缺陷。

### 测试新构建（Test New Build）

测试应用程序的新构建，重复上面的过程，直到缺陷被修复。

### 分析缺陷数据（Analyze Defect Data）

产生报告和图表来帮助我们分析缺陷修复过程，并帮助我们决定什么时候发布该产品。

#### 3.4.2 TestDirector 安装

TestDirector 是一种 B/S 结构的测试管理工具，它利用 IIS 作为服务器，Sql Server 或者 Oracle 作为系统数据库，用户通过浏览器进行访问使用。可以安装部署在 Windows2000, XP, 2003 等系统上，但一般情况下不在 Windows XP 系统上安装 web 服务器，因为 XP 系统本身有一个连接数的限制（10 个连接），这样对系统服务来说是一个瓶颈，无法达到使用的要求，故一般的 Web 服务需要安装在服务器版的操作系统上，如 Windows 2000 Server、Windows 2003 Server 等。所以，TestDirector 需要部署在带有服务功能的系统上，比如 Windows2000 server, Windows 2003 server 等。

一般系统支持服务情况如表 3-1 所示：

操作系统	Web 服务支持	数据库支持
Windows 2000 Server	IIS 、 TOMCAT 、 JBOSS 、 WEBLOGIC 等	MYSQL 、 SQL SERVER 、 ORACLE 等
Windows 2000 Advance Server	IIS 、 TOMCAT 、 JBOSS 、 WEBLOGIC 等	MYSQL 、 SQL SERVER 、 ORACLE 等
Windows 2000 Professional	IIS 、 TOMCAT 、 JBOSS 、 WEBLOGIC 等	MYSQL 、 SQL SERVER 、 ORACLE 等
Windows 2003 Server	IIS 、 TOMCAT 、 JBOSS 、 WEBLOGIC 等	MYSQL 、 SQL SERVER 、 ORACLE 等
Windows XP	IIS 、 TOMCAT 、 JBOSS 、 WEBLOGIC 等	MYSQL、ORACLE 等

表 3-1 Windows 系统支持服务类型

这里我们将 TestDirector 安装部署在 Windows2003 系统上，数据库使用 Sql Server 2000 中文企业版。

在安装 TestDirector 之前，我们先进行 Sql Server 2000 中文企业版的安装。

## Sql Server 2000 数据库安装

我们这里采用 Sql Server 2000 作为我们 TestDirector 的后台数据库，下面简单介绍一下如何在 Windows 2003 Server 系统中安装 Sql Server 2000 数据库。

注意：

在安装 Sql Server 2000 之前，确保操作系统已经为超级管理员（administrator）设置了密码。

如果安装程序是光盘的话，我们只需将安装光盘插入光驱即可，如果是 ISO 镜像文件，则使用虚拟光驱。

在安装之前，我们先了解一下 SQL SERVER 2000 版本与安装要求。

SQL Server 2000 有七种不同版本。它们是：企业版、标准版、个人版、开发版、Microsoft Windows CE 版、企业评估版和桌面引擎（MSDE 2000）。

SQL Server 2000 对操作系统的软件要求如下：

SQL Server 组件	受支持的 Microsoft Windows 版本
企业版	Microsoft Windows NT Server 4.0 Service Pack 5 或更高版本、 Windows NT Server 4.0 Enterprise Edition Service Pack 5 或更高版本、 Windows 2000 Server、 Windows 2000 Advanced Server 或更高版本
标准版	Microsoft Windows NT Server 4.0 Service Pack 5 或更高版本、 Windows NT Server 4.0 Enterprise Edition Service Pack 5 或更高版本、 Windows 2000 Server、 Windows 2000 Advanced Server 或更高版本
个人版	Windows XP Home、 Windows XP Professional、 Microsoft Windows 98、 Windows Me、 Windows NT Workstation 4.0 Service Pack 5 或更高版本、 Windows NT Server 4.0 Service Pack 5 或更高版本、 Windows NT Server 4.0 Enterprise Edition Service Pack 5 或更高版本、 Windows 2000 Professional、 Windows 2000 Server、

	Windows 2000 Advanced Server 或更高版本
开发版	Windows XP Home、 Windows XP Professional、 Microsoft Windows NT Workstation 4.0 Service Pack 5 或更高版本、 Windows NT Server 4.0 Service Pack 5 或更高版本、 Windows NT Server 4.0 Enterprise Edition Service Pack 5 或更高版本、 Windows 2000 Professional、 Windows 2000 Server、 Windows 2000 Advanced Server 或更高版本

表 3- 2 SQL Server 2000 对操作系统要求

SQL Server 2000 对硬件的要求如下：

硬件	最低要求
CPU	Intel 或兼容的 Pentium 166 MHz 或更快。
内存 (RAM)	最少 64 MB，建议使用 128 MB 或更多（如果操作系统有更高要求，要先满足）
硬盘驱动器空间	约 400 MB
显示器	VGA 或更高分辨率。需要 800x600 或更高分辨率用于 SQL Server 图形工具
指针设备	Microsoft 鼠标或兼容的鼠标
CD-ROM 驱动器	必需

表 3- 3 SQL Server 2000 对硬件要求

在了解了安装要求后，我们开始在 Windows 2003 Server 系统上安装 Sql Server 2000 。

1、启动 Windows 2003 Server 系统，Windows 2003 Server 系统最好打上最新的补丁文件。

2、放入光盘，或加载虚拟光驱。如果是镜像文件（ISO 文件），不要解压，使用虚拟光驱，如 DAEMON Tools。

3、选择【安装 SQL Server 2000 组件】。



图 3-46 安装 SQL Server 2000 组件

4、选择【安装数据库服务器】。



图 3-47 安装数据库服务器

5、点击【下一步】。

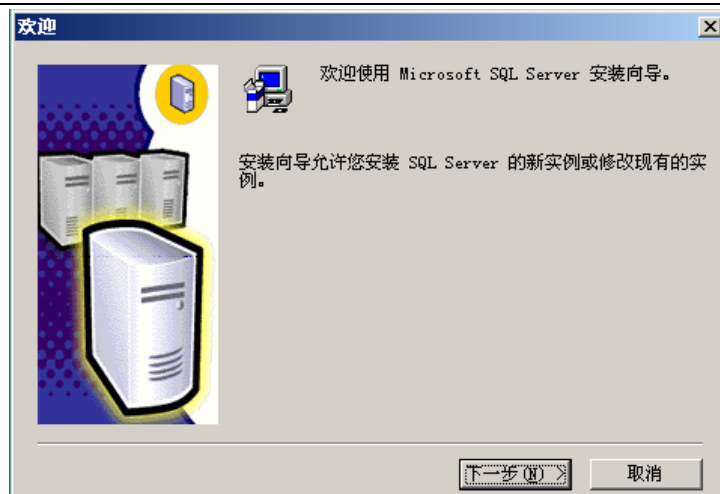


图 3-48 使用安装向导

6、默认，点击【下一步】。



图 3-49 选择本地计算机

7、默认，点击【下一步】。

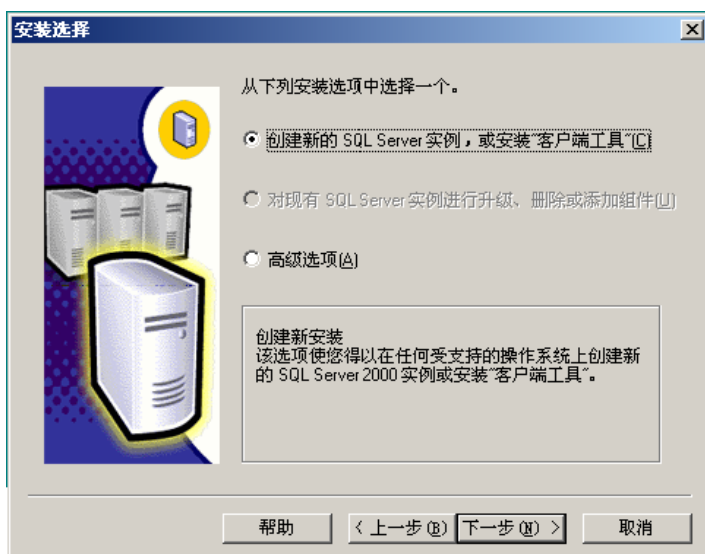


图 3- 50 创建数据库实例

8、输入姓名与公司名称，点击【下一步】。



图 3- 51 填写用户信息

9、选择“是”。

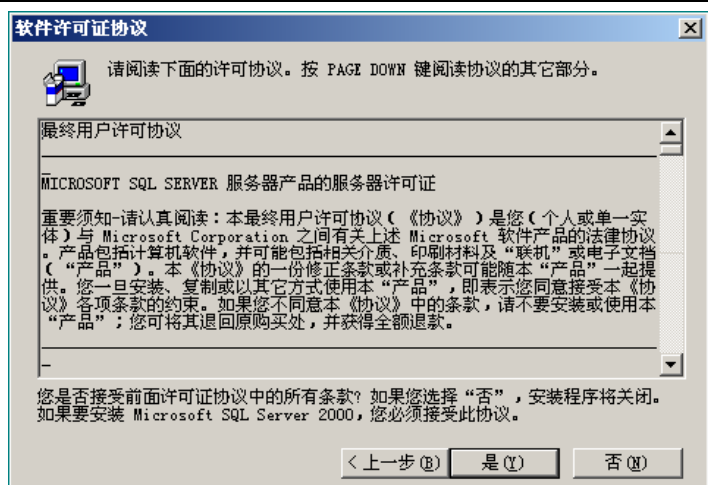


图 3-52 软件许可证协议

10、选择服务器和客户端工具，点击【下一步】。



图 3-53 选择服务器和客户端工具

11、默认，点击【下一步】。



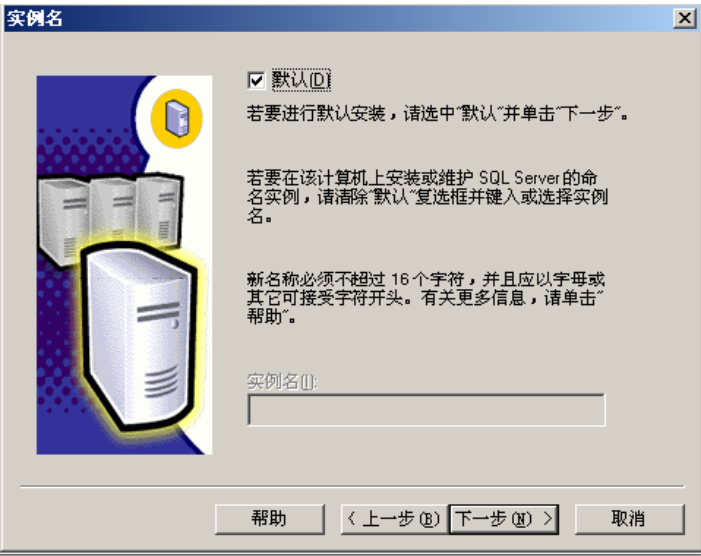


图 3- 54 设置实例名

12、默认，点击【下一步】。

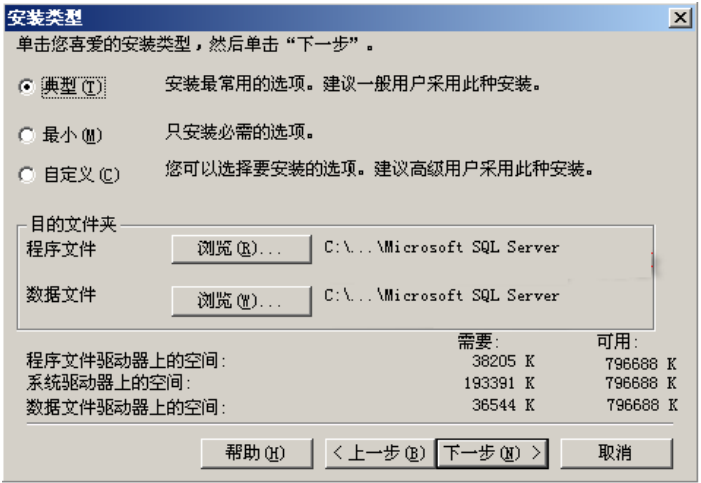


图 3- 55 选择安装类型

13、选择“使用本地系统帐户(L)”,点击【下一步】。

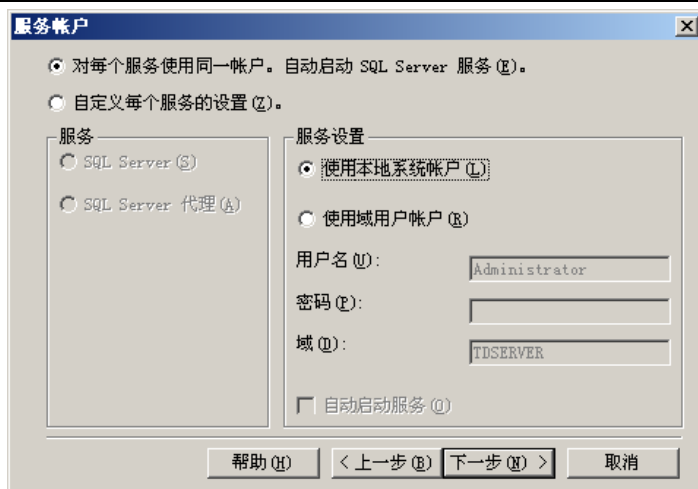


图 3-56 设置服务帐户

14、按图 3-57 所示选择，输入密码后点击【下一步】。

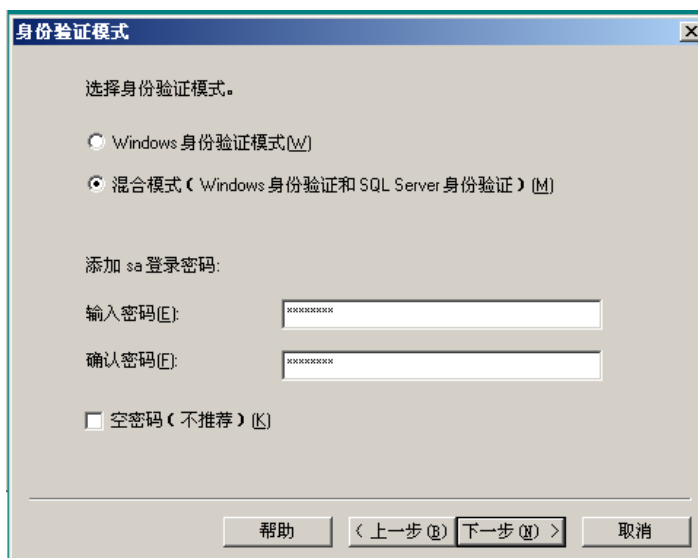


图 3-57 设置身份验证模式

15、点击【下一步】。

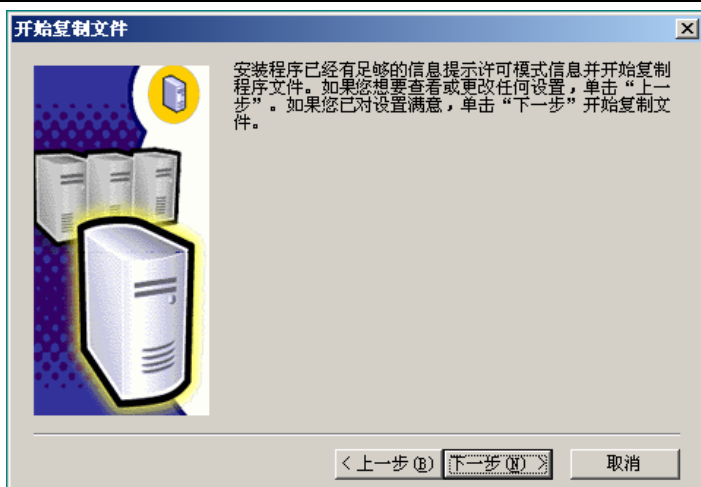


图 3-58 开始复制文件

16、按图 3-59 所示输入，点击【继续】。

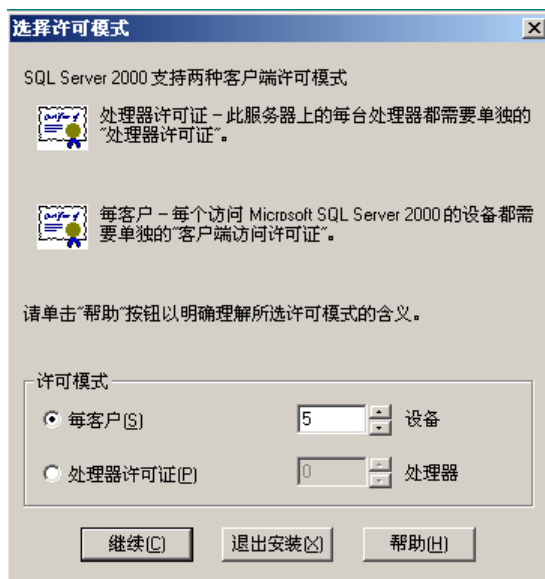


图 3-59 选择许可模式

17、安装过程进行中。

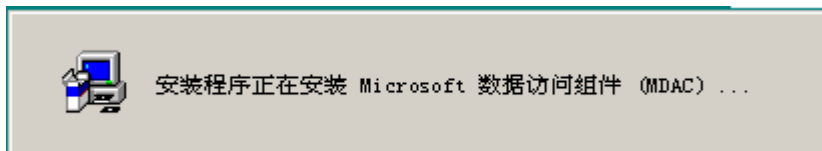


图 3-60 安装进程

18、安装过程进行中。

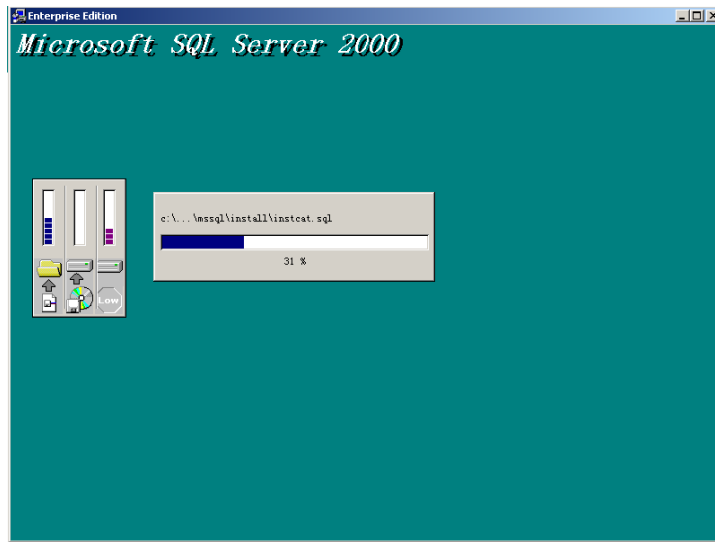


图 3- 61 安装进程

19、打补丁，给数据库打上 sp4 补丁，解压补丁文件。



图 3- 62 解压补丁文件

20、解压进行中。

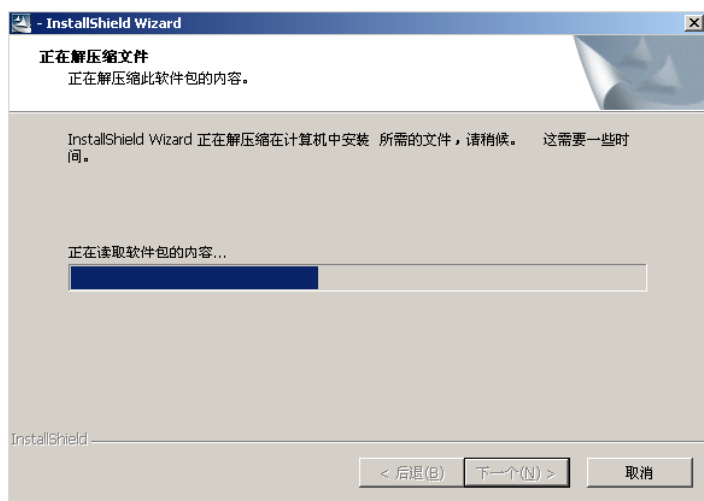


图 3- 63 解压补丁文件进程

21、解压完成。

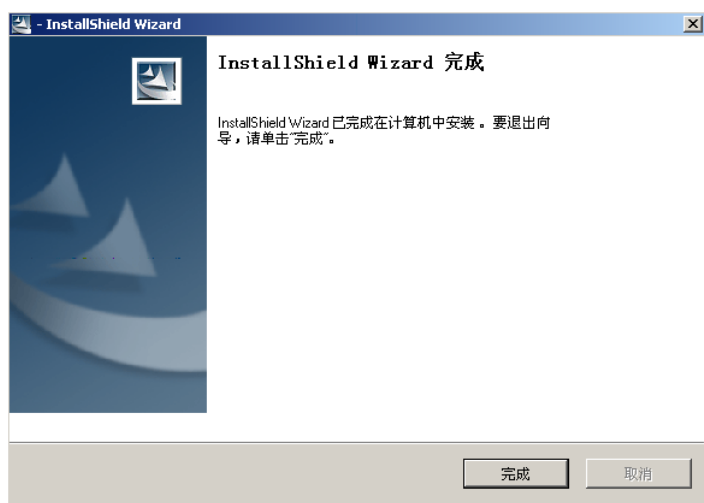


图 3- 64 解压完成界面

22、打开解压后的文件目录，文件名为“SQL2KSP4”。

23、点击“setup”，运行安装文件。

24、点击【下一步】。

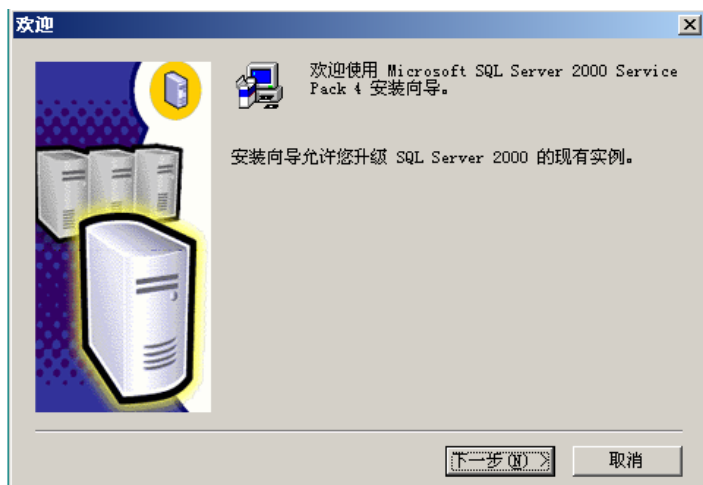


图 3-65 补丁安装界面

25、点击【是】。

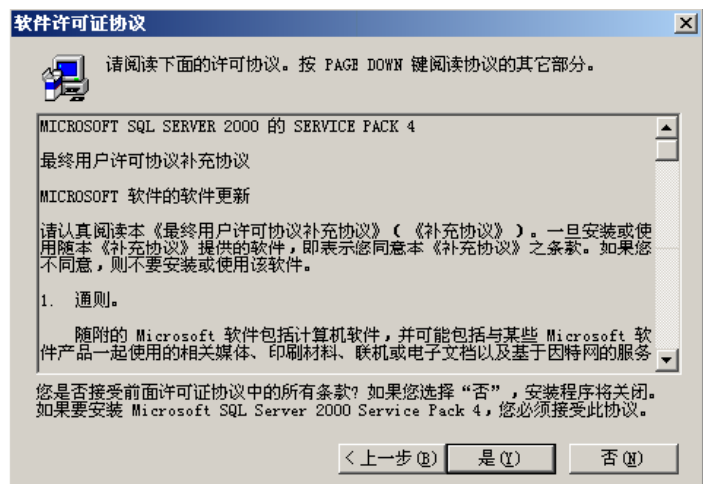


图 3-66 补丁许可证协议

26、点击【下一步】。

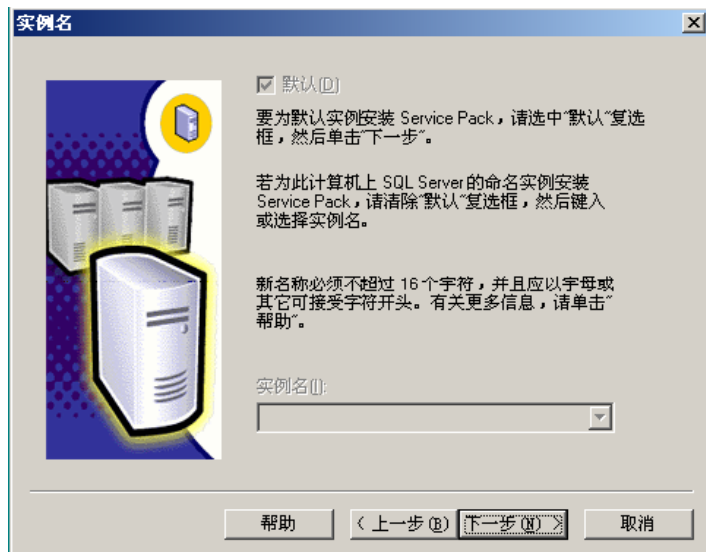


图 3-67 选择实例名

27、按图 3-68 所示选择, 点击【下一步】。

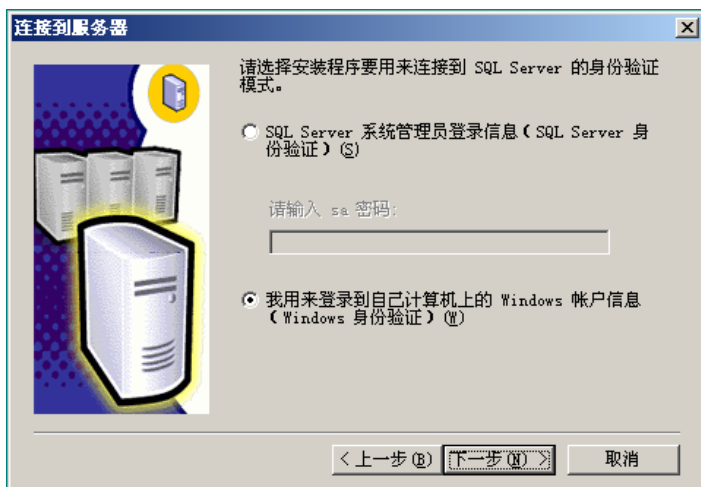


图 3-68 设置连接服务器帐号

28、安装进行中。

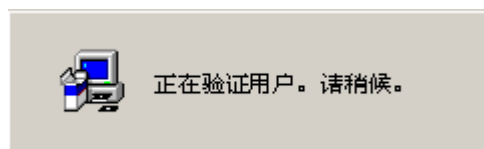


图 3-69 验证安装用户

29、按图 3- 70 选择，点击【继续】。

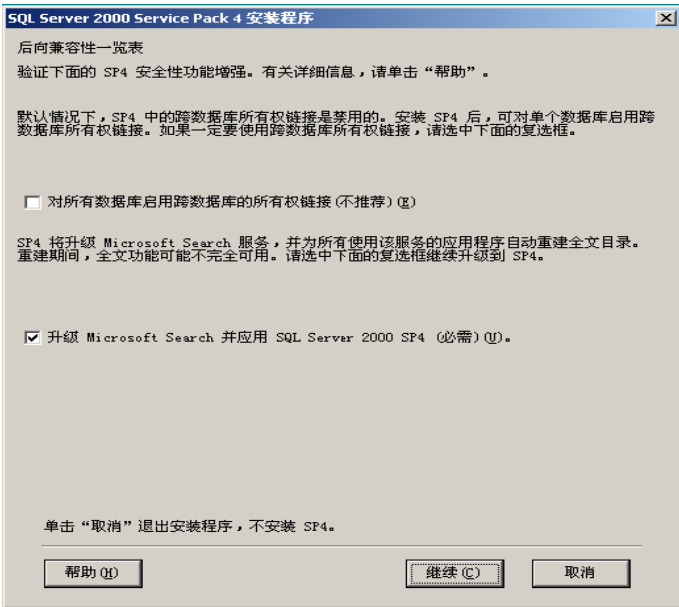


图 3- 70 选择升级选项

30、点击【确定】。

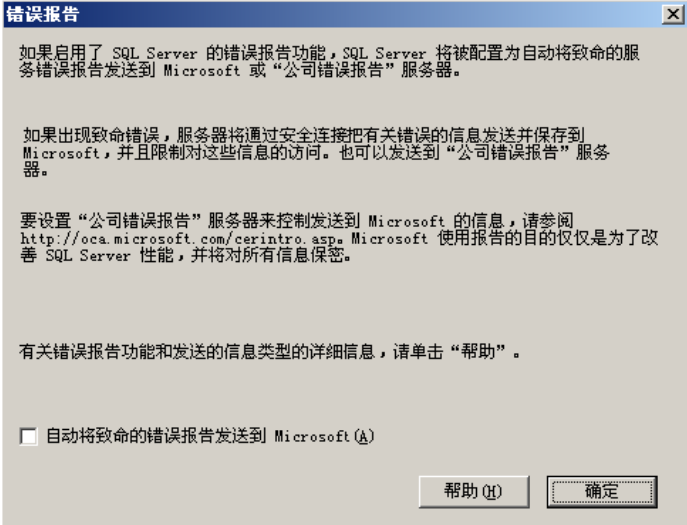


图 3- 71 错误报告设置

31、点击【下一步】。



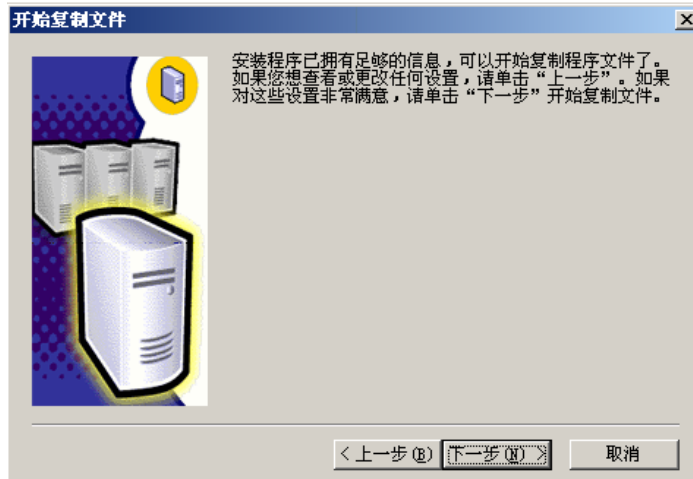


图 3-72 开始复制文件

32、安装进行中。

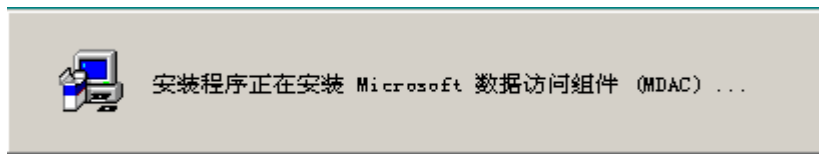


图 3-73 安装进程

33、安装进行中。

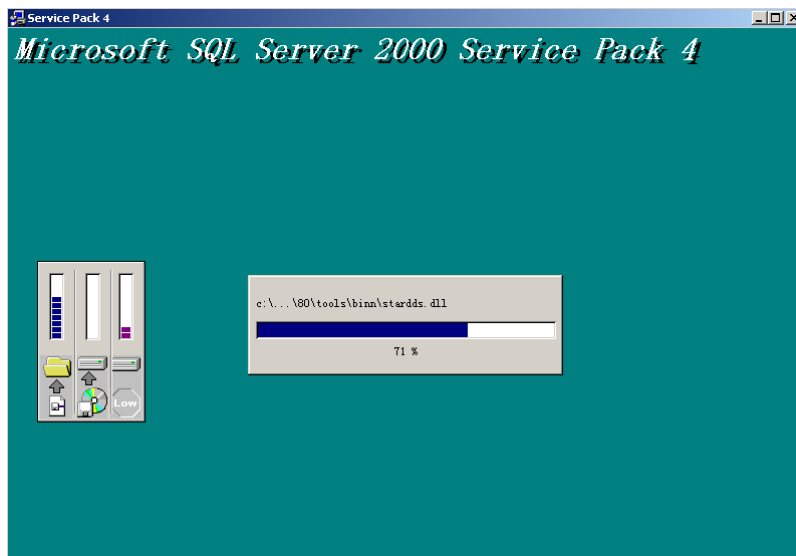


图 3-74 安装进程

34、安装进行中。

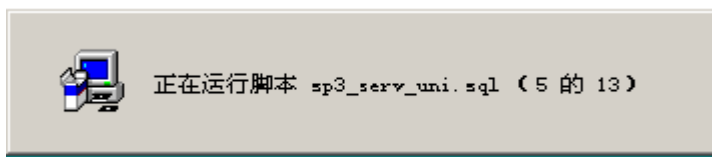


图 3-75 运行安装 sql 脚本

35、安装完成。



图 3-76 补丁安装完成

36、重启电脑，完成数据库安装。

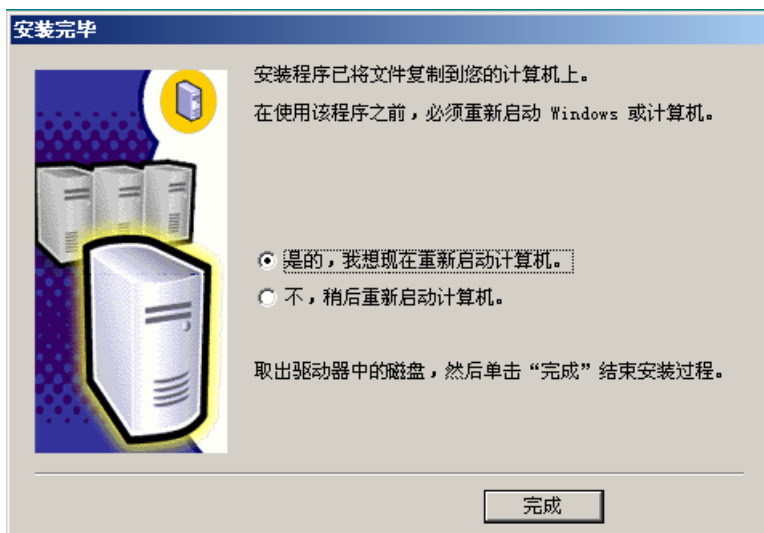


图 3-77 安装完毕重启

至此，Sql Server 2000 在 Windows 2003 Server 系统安装完成。

#### TestDirector 安装

Sql Server 2000 在 Windows 2003 Server 系统安装完成后，我们开始 TestDirector 的安装，具体安装步骤如下：

- 1、点击 Setup.exe 运行安装程序。
- 2、点击【Next】。



图 3-78 TestDirector 安装界面

- 3、输入 License，如果没有，可选中“Use Evaluation key”，享受大约 60 天的试用期。我们这里选中该项。点击【Next】。

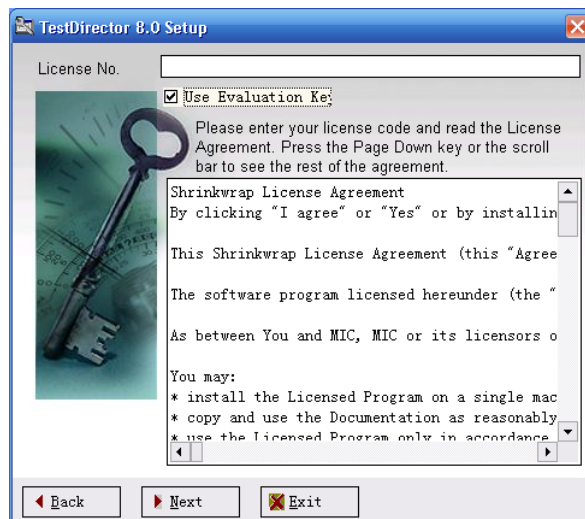


图 3-79 输入 License No

- 4、数据库选择区我们使用前面安装的 SQL Server 2000,故这里需勾中 MS-SQL SERVER,点击【Next】。

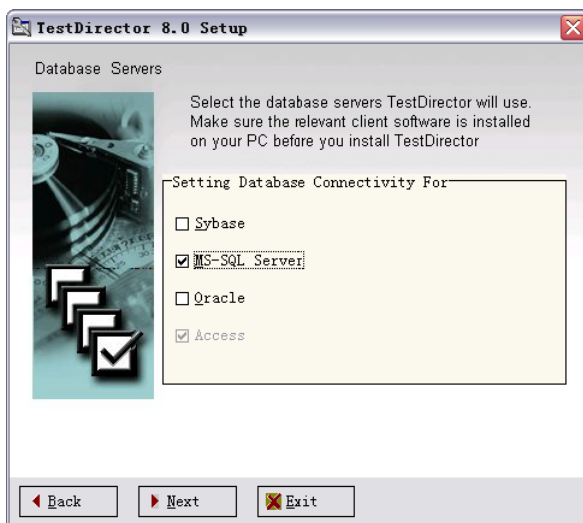


图 3- 80 选择数据库

5、默认即可，点击【Next】

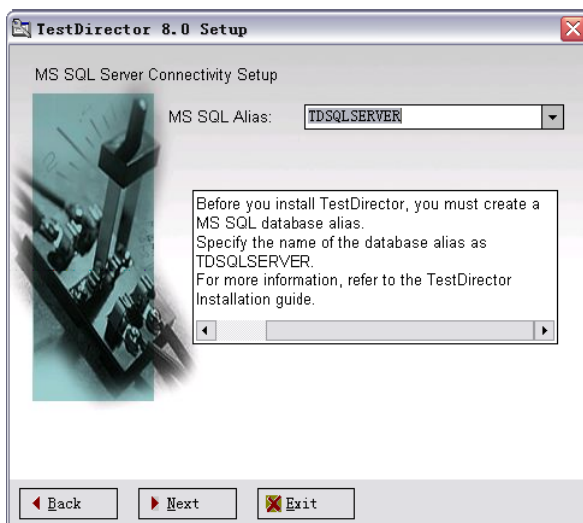


图 3- 81 设置数据库服务器

选择数据库服务器，如果数据库不是安装在本机，则需“MS SQL Alias:”的下列表框中重新选择一个，这里我们默认即可。

6、输入系统超级用户密码，点击【Next】



图 3- 82 设置 TestDirector 服务器管理员

此处是使 TestDirector 拥有系统管理员的权限，并将 TestDirector 注册为一个系统服务。要求系统管理员必须有密码，所以为什么前面我们一直强调，安装完毕系统后必须给系统管理员（administrator）设置一个密码。

7、默认即可，点击【Next】。

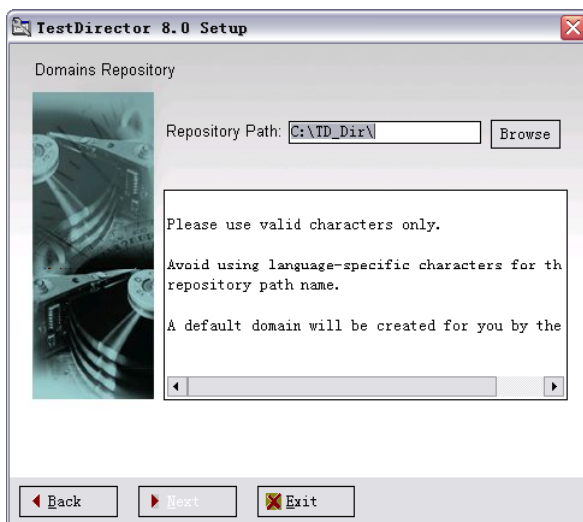


图 3- 83 设置域文件路径

8、询问是否创建 TestDirector\_Dir 文件夹，点击【Yes】。

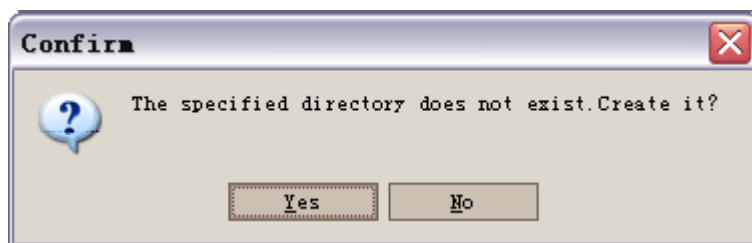


图 3-84 创建域文件目录

9、询问是否共享 TestDirector\_Dir，点击【Yes】。

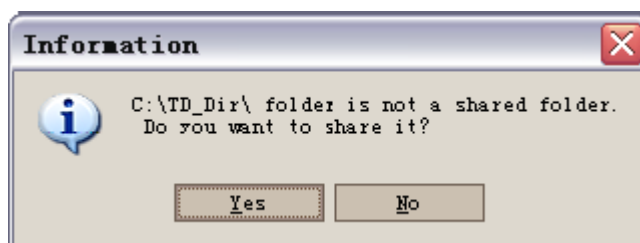


图 3-85 将域文件设为共享

10、设置共享名称，默认即可，点击【OK】。

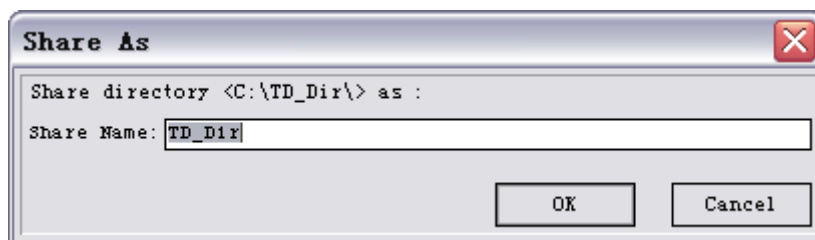


图 3-86 设置共享名

11、配置邮件服务，如果有 SMTP 服务器则可输入，没有就任意输一个，点击【Next】。

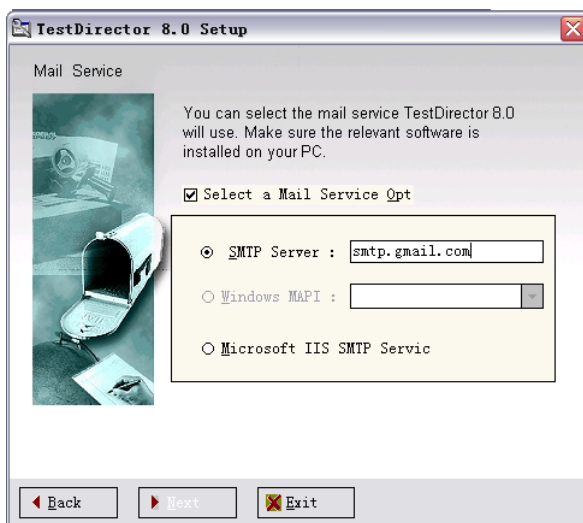


图 3-87 设置 SMTP 服务器

12、创建 IIS 虚拟目录，默认即可，点击【Next】。

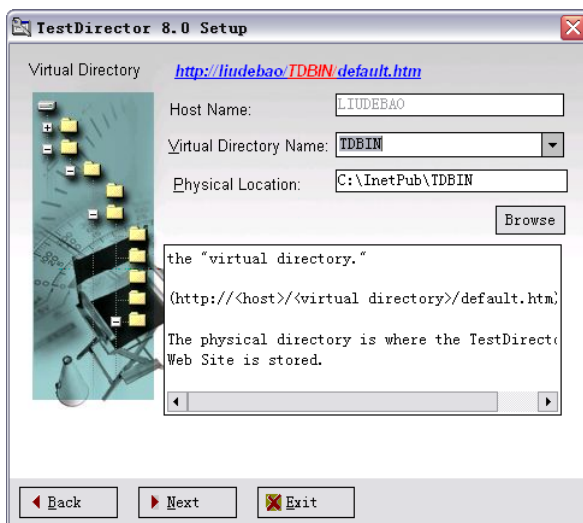


图 3-88 创建 IIS 目录

13、询问目录不存在，是否创建，点击【Yes】。

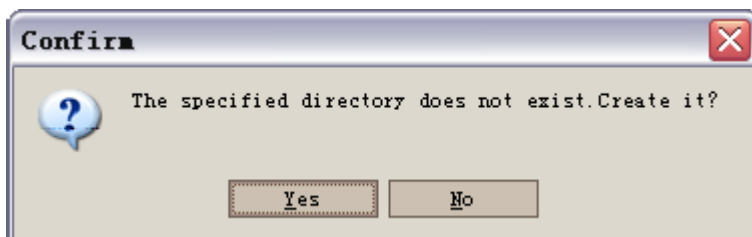


图 3- 89 创建 IIS 目录

14、询问是否安装 SiteScope，不安装，默认设置，点击【Next】。

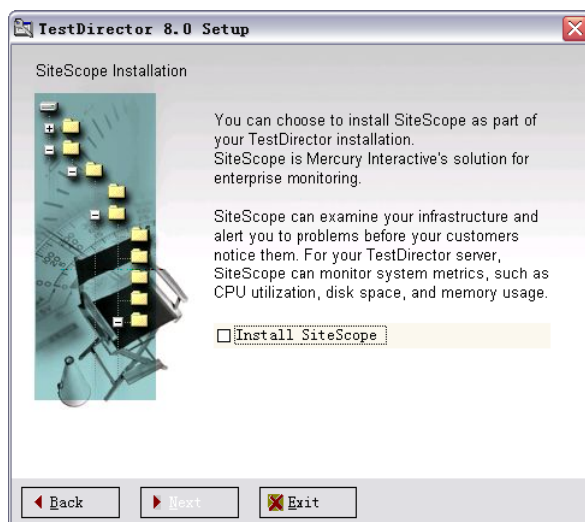


图 3- 90 Install SiteScope 选项界面

15、默认即可，点击【Next】。

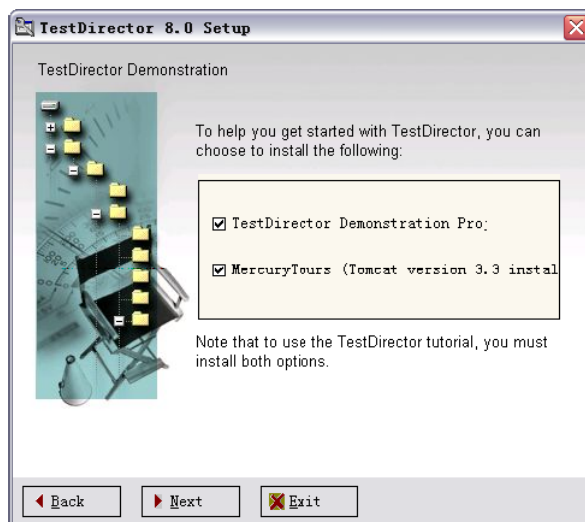


图 3- 91 TestDirector 演示系统选择界面

16、点击【Install】，执行安装。



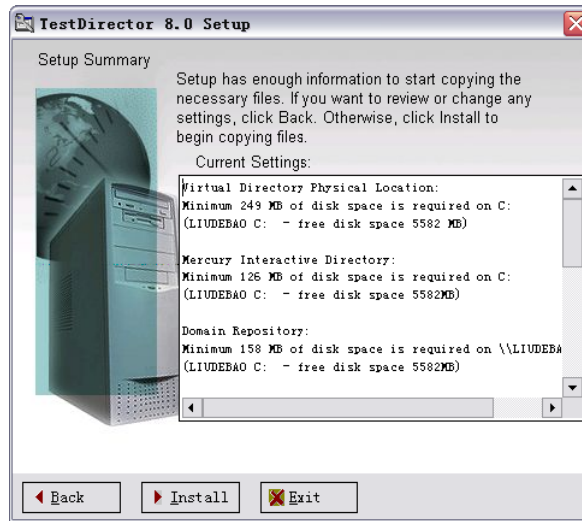


图 3- 92 TestDirector 安装信息

17、安装进行中。

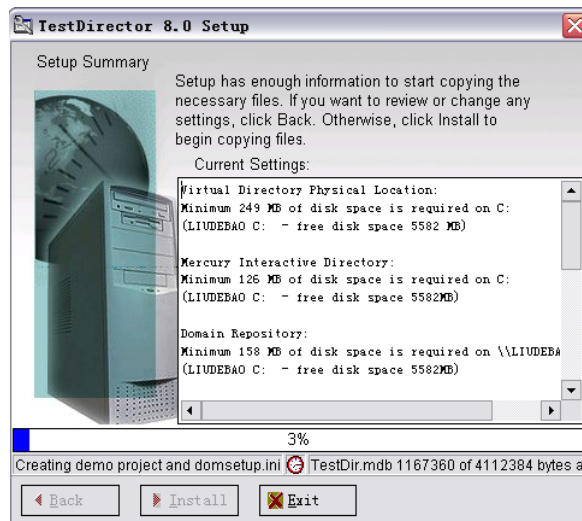


图 3- 93 TestDirector 安装进程

18、点击【Next】。



图 3- 94 TestDirector 注册

19、点击【Yes】，完成安装。

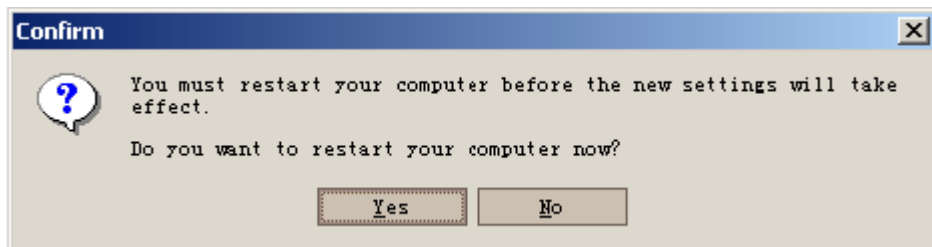


图 3- 95 安装重启选择界面

20、打补丁 TestDirector80SP2，直接点击“TestDirector80SP2”即可安装。

21、补丁安装过程中，要求输入操作系统管理员（administrator）密码，输入后点击【OK】。



图 3-96 TestDirector 管理员密码输入界面

22、重启电脑，完成安装。

至此，TestDirector 服务器安装完成，在客户端（任意局域网内其它机器）利用 IE 访问 <http://IP/TDBIN>，检查是否安装成功。

小提示：

此处的 IP 是 TestDirector 服务器的 IP,也可使用 TestDirector 服务器的机器名，初始管理员用户密码为空。

### 3.4.3 TestDirector 后台管理

正确安装好 TestDirector 后，我们就可以利用 TestDirector 管理我们的测试工作了，在使用之前，先花点时间来熟悉一下一般测试管理的思路。

我们知道，测试项目的管理需根据不同的项目制定不同管理的方法，在实际的实施过程中会结合项目的实际情况进行调整。比如测试人员的安排、项目工作小组的配置，以及工作流程的设定等等。在此之前，我们需要建立一个项目，来包含我们所要进行的管理工作。那么一般的项目管理思路如 图 3-97 所示：

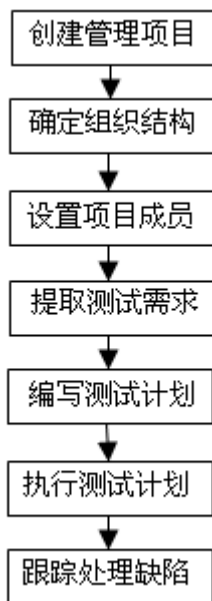


图 3- 97 项目管理流程图

就像 图 3- 97 所示，要管理测试项目，就需要先在 TestDirector 建立一个对应的项目，然后这个基础上再开展测试管理工作。

TestDirector 后台管理功能提供了创建项目的功能，这里我们按照下面的步骤使用 TestDirector 后台管理功能并创建测试项目。

### TestDirector 后台登录

1、浏览器中输入 <http://192.168.0.102/tdbin/SiteAdmin.htm>，进入 TestDirector 系统。

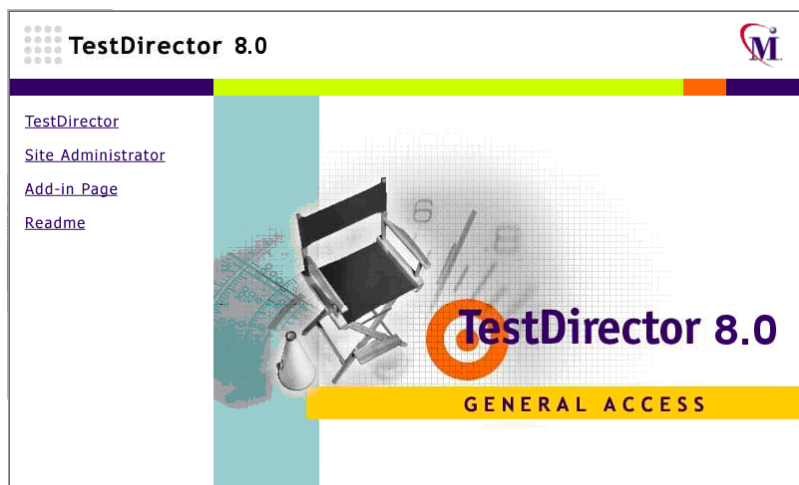


图 3- 98 TestDirector 使用界面

192.168.0.102 为 TestDirector 服务器地址，可根据实际情况调整。

2、点击 Site Administrator，进入 图 3-99 所示页面。



图 3-99 TestDirector 后台登录界面

3、初始的管理员密码为空，点击“Login”，进入管理页面。

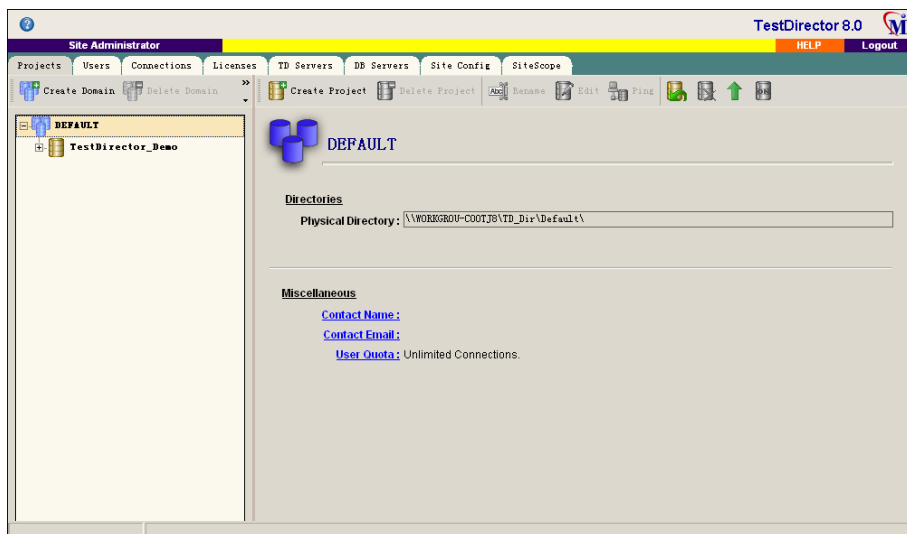


图 3-100 后台管理界面

TestDirector 后台管理功能我们主要学习三种：数据库连接维护；项目维护；用户维护。其它功能请学者自行研究。

### 数据库连接维护

在管理界面选择 DB Servers，进入数据库维护界面，如 图 3-101 所示。

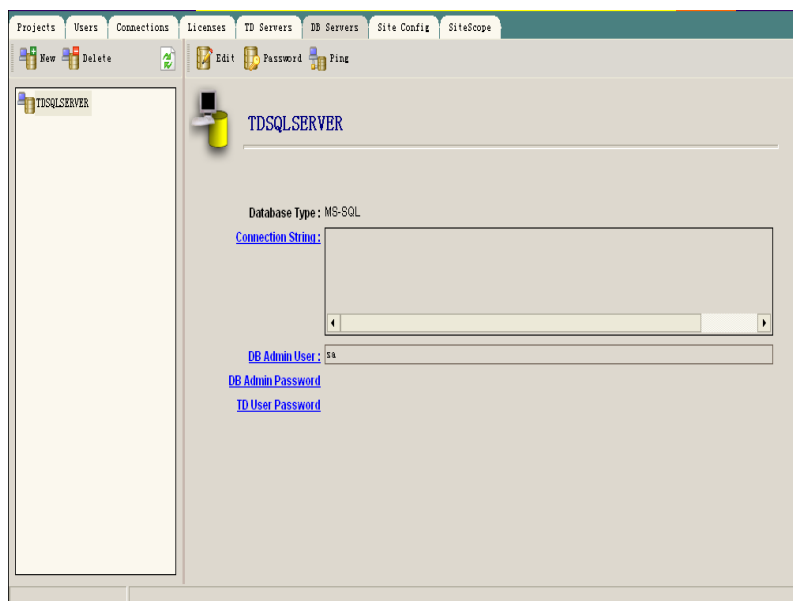


图 3- 101 数据库维护界面

我们在安装 TestDirector 时，已经选择了 Sql Server 数据库，所以这里会有一个默认的数据库服务器。这里需要设置一下该数据库的配置信息。

### 新建数据库服务

- 1、点击【New】，打开新建窗口。



图 3- 102 新建功能窗口

- 2、输入相关创建信息，然后点击【OK】。

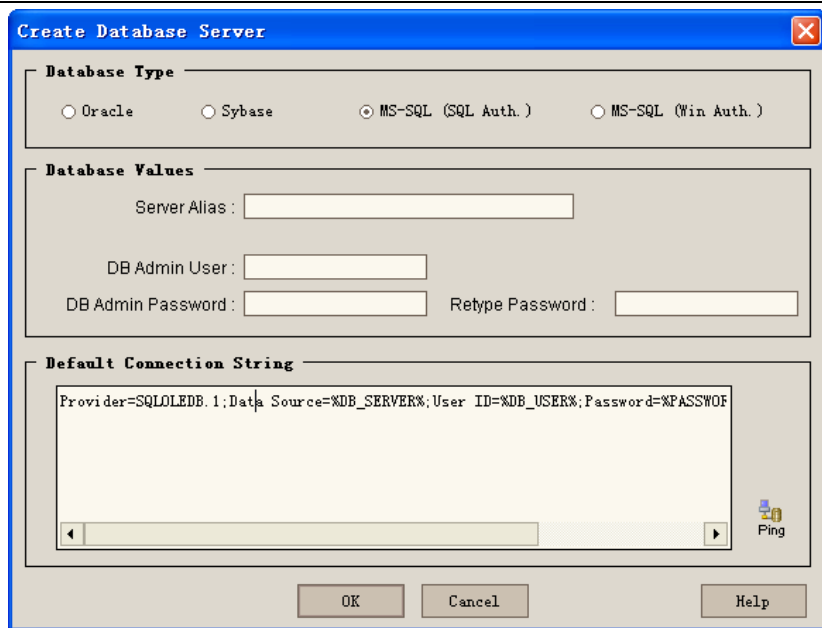


图 3- 103 创建数据库服务器

Database Type 选择 MS-SQL (SQL Auth)，使用 SQL 的用户验证，Database Values 下的 Server Alias 处输入数据库服务器的 ip，或者服务名，DB Admin User 输入数据库连接用户名，一般为 sa，DB Admin Password 输入数据库连接密码，Retype Password 输入同样的密码。

完成后点击 Ping，检查设置是否正确，如果显示 图 3- 104 则表示设置成功。

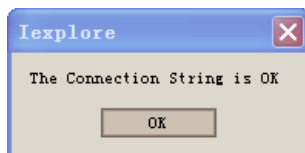


图 3- 104 连接成功

3、成功界面。

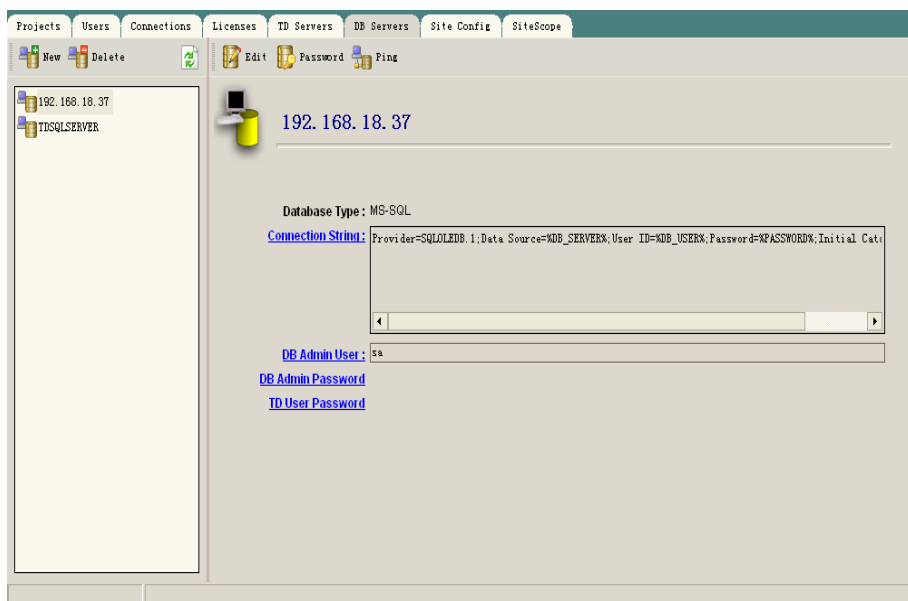


图 3- 105 数据库服务器信息

## 维护数据库服务

- 1、选中左边的数据库服务，点击Edit，进入编辑界面，如 图 3- 106 所示。

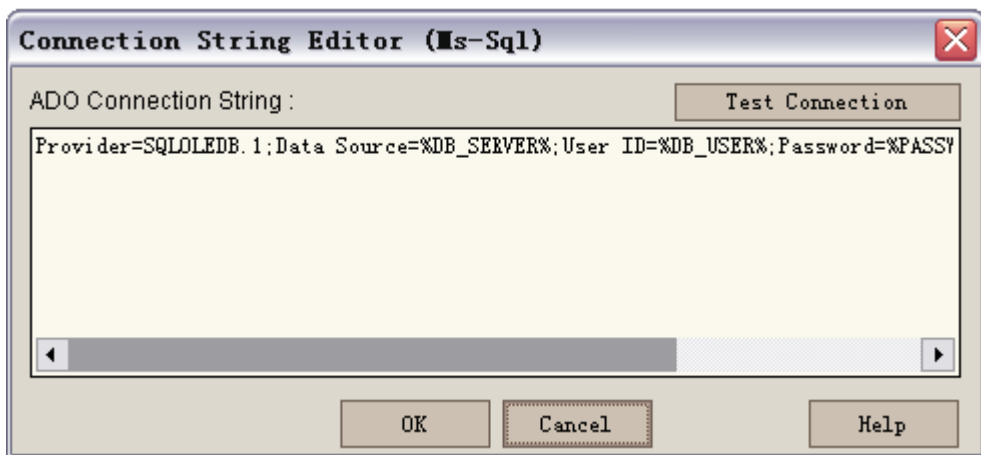


图 3- 106 编辑数据库连接串

- 2、修改 Data Source。

如果使用本机 SQL Server，则将“Data Source=%DB\_SERVER%”改为“Data Source=127.0.0.1”。

如果使用其它机器上的 SQL Server，则将“Data Source=%DB\_SERVER%”改为“Data Source=其它机器 IP”。



### 3、测试连接。

点击 “Test Connection”，输入数据库密码，确定后，会显示连接成功。如不成功，请检查数据源设置。

### 4、修改数据库连接密码。

点击“Password”，输入旧密码与新密码，点击确定，完成后会显示 图 3- 107。

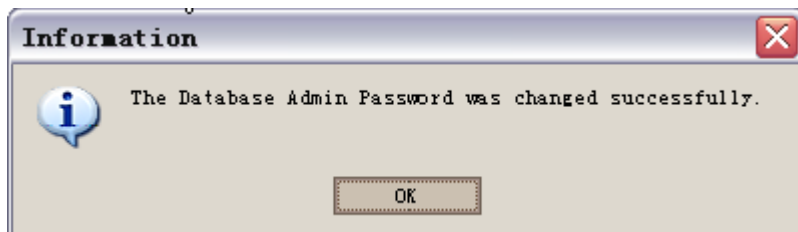


图 3- 107 修改数据库管理员密码

### 5、Ping 数据源，检查连接是否正常。

点击“Ping”，输入数据库用户名与密码，确定后成功 ping 通，如不通请查找原因。

## 删除数据库服务

1、选中需删除的数据库服务，点击【Delete】。

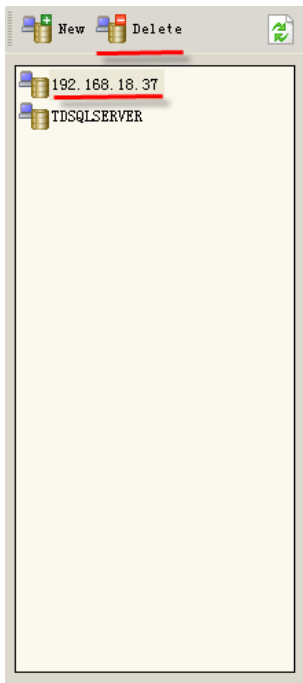


图 3- 108 删除数据库服务器

2、询问是否确定需要删除该数据库服务，点击【Yes】。



图 3- 109 删除数据库服务器确认信息

3、删除成功后，该数据库服务将不再显示。

### 项目维护

在管理界面选择Projects，进入项目维护界面，如 图 3- 110。

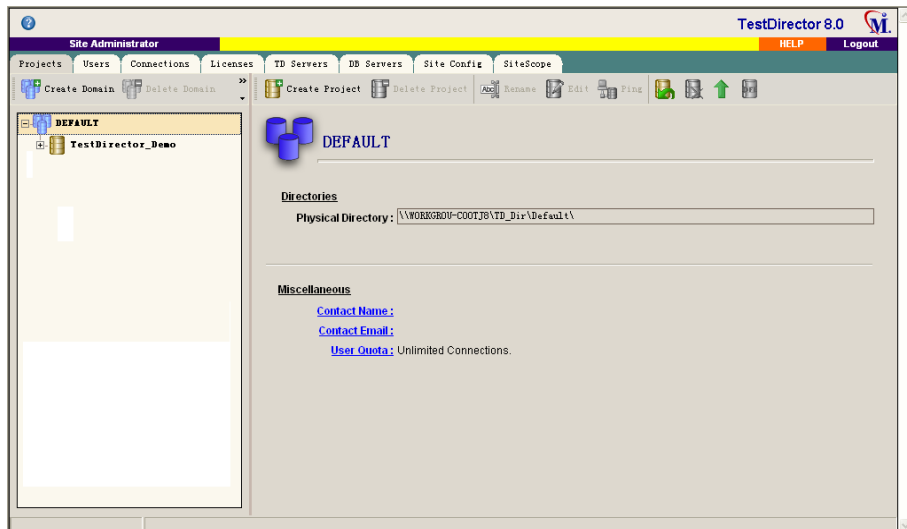


图 3- 110 项目维护界面

### 新建项目

1、在界面中点击“Create Project”。



图 3- 111 创建项目

2、输入项目名称并选择数据库类型，点击【Next】。

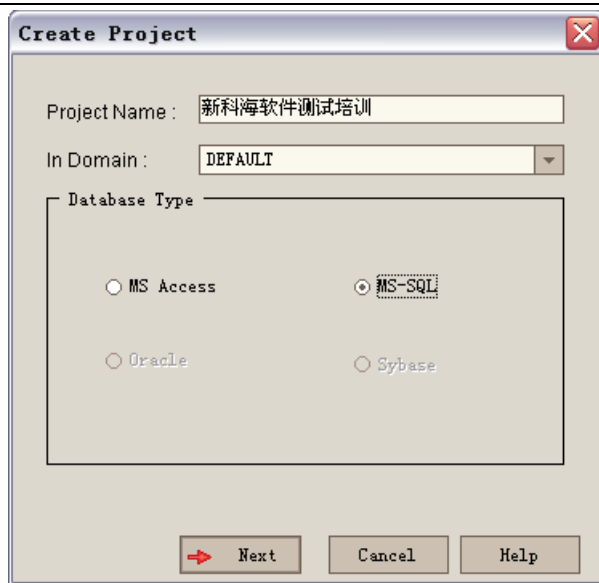


图 3- 112 创建项目信息

如果使用的是其它的数据库，则可选择相对应的数据库，这里我们使用的是 MS-SQL。

3、选择数据库，并输入用户名与密码，点击【Next】。

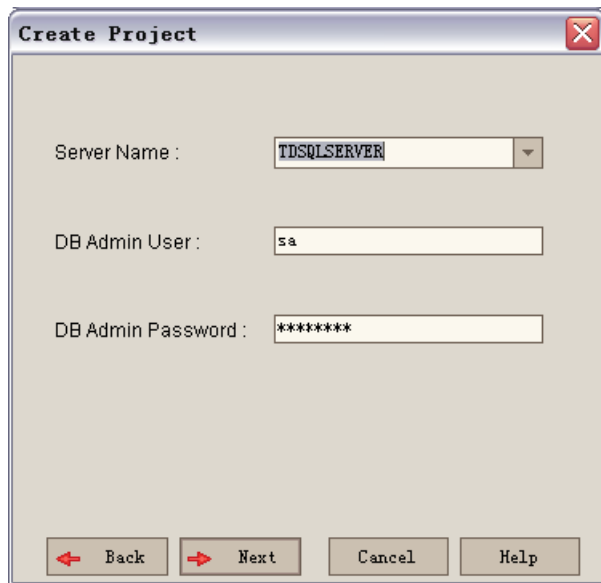


图 3- 113 选择数据库服务器

4、检查无误后，点击【Create】。

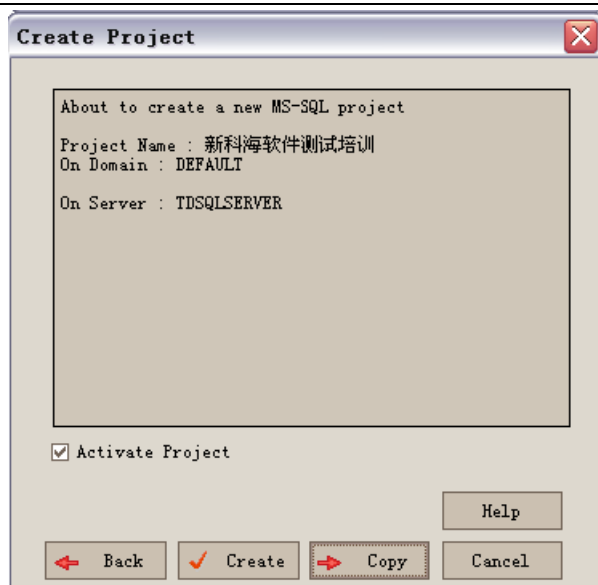


图 3- 114 创建项目

5、成功创建项目。

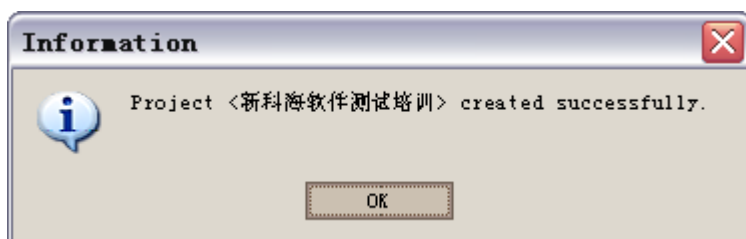


图 3- 115 项目创建成功信息

## 项目维护

1、点击 Rename，出现 图 3- 116。

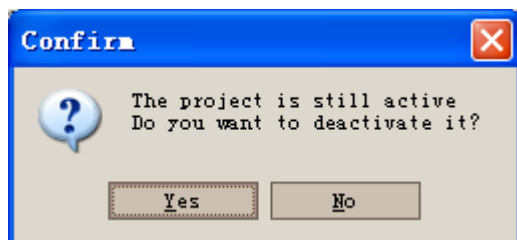


图 3- 116 暂停项目确认信息

此处描述的意思是说当前的项目正在使用过程中，是否需要停止使用，点击【Yes】。

2、断开该项目所有用户连接。

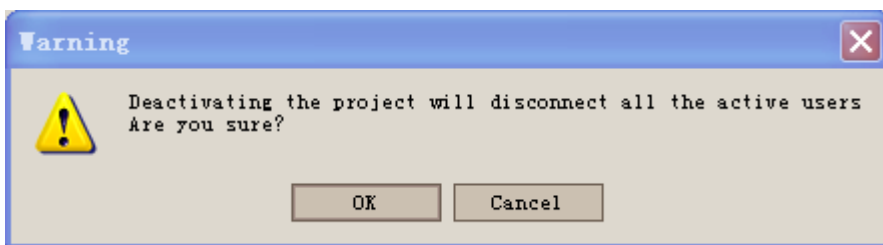


图 3- 117 断开所有用户连接确认信息

此处意思是说停止该项目将会断开所有活动用户的连接，点击【OK】。

3、输入新的项目名，点击【OK】。项目停止后，项目的状态为红色，并且标注为“Not Active”。

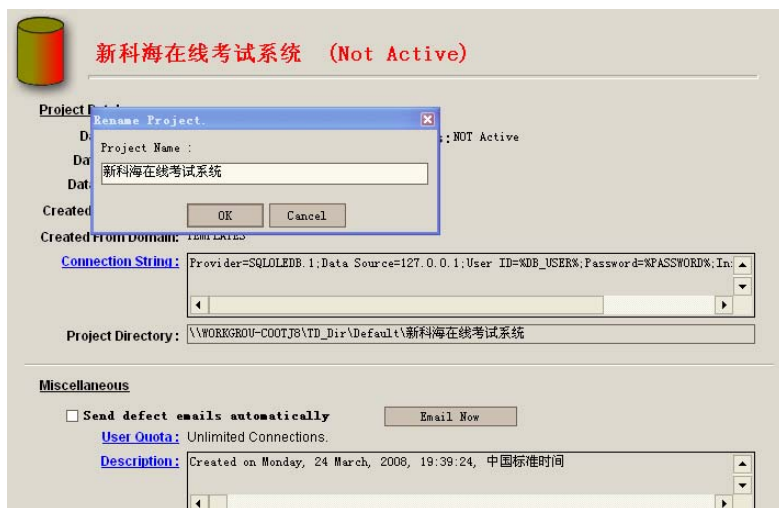


图 3- 118 重命名项目

4、启用重命名后的项目，选中重命名后的项目，点击右键，选择“Activate Project”，完成启用。

5、修改该项目数据库连接信息。点击“Edit”，出现图 3- 119。

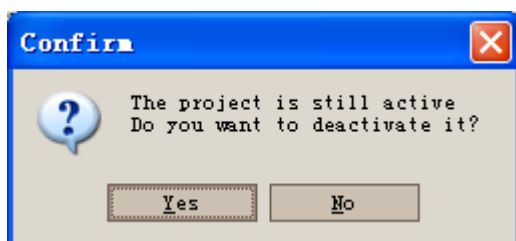


图 3-119 暂停项目确认信息

此处描述的意思是说当前的项目正在使用过程中，是否需要停止使用，点击【Yes】。

6、断开该项目所有用户连接。

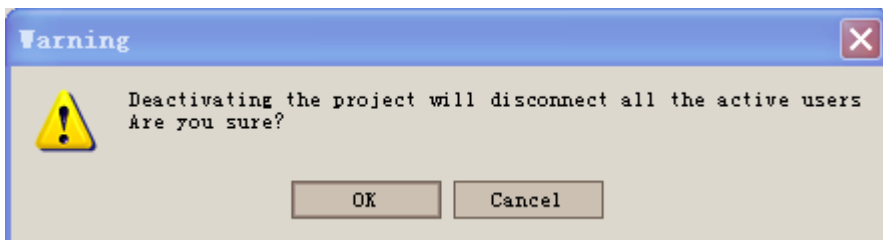


图 3-120 断开活动用户连接确认信息

此处意思是说停止该项目将会断开所有活动用户的连接。点击【OK】。

7、修改数据库配置信息。方法与前面的维护数据库服务一样。

8、启用维护后的项目，方法与前面重命名后启用项目一样。

## 删除项目

删除项目就比较简单了，参照下面的步骤进行。

1、选中需删除的项目，点击

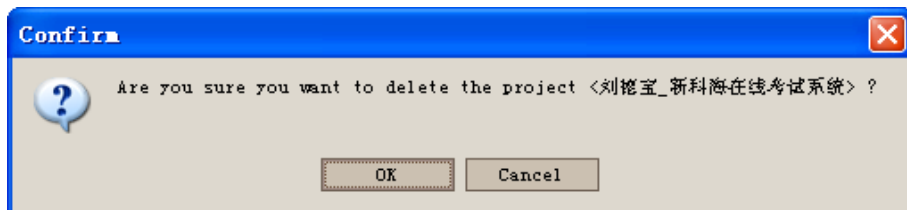


图 3-121 删除项目确认信息

询问是否真的要删除该项目，点击【OK】。

2、询问是否停用该项目。

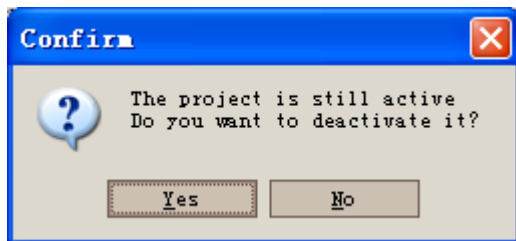


图 3- 122 暂停项目确认信息

此处描述的意思是说当前的项目正在使用过程中，是否需要停止使用，点击【Yes】。

3、断开该项目所有用户连接。

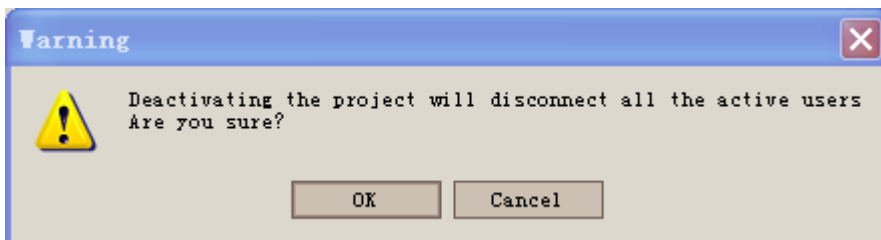


图 3- 123 断开活动用户连接确认信息

此处意思是说停止该项目将会断开所有活动用户的连接。点击【OK】。

4、输入数据库用户名和密码，点击【OK】。



图 3- 124 输入数据库服务器用户名与密码

5、删除完成。

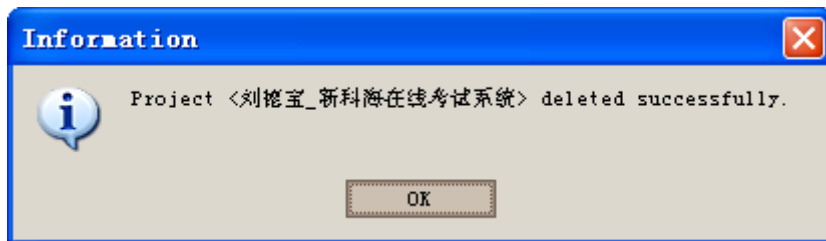


图 3- 125 项目删除成功信息

注意：

所有的修改、删除操作对于正在使用过程中的项目是不可操作的，必须将这些项目停用后才能进行相关的操作，在修改，删除前请慎重考虑这些改变是否必要

用户维护

用户在整个测试管理过程中是非常重要的。TestDirector 后台提供了新增，修改，删除系统用户的功能。

此处的用户是针对整个管理系统的，不区分项目。

在管理界面选择Users，进入项目维护界面，如图 3- 126 所示。

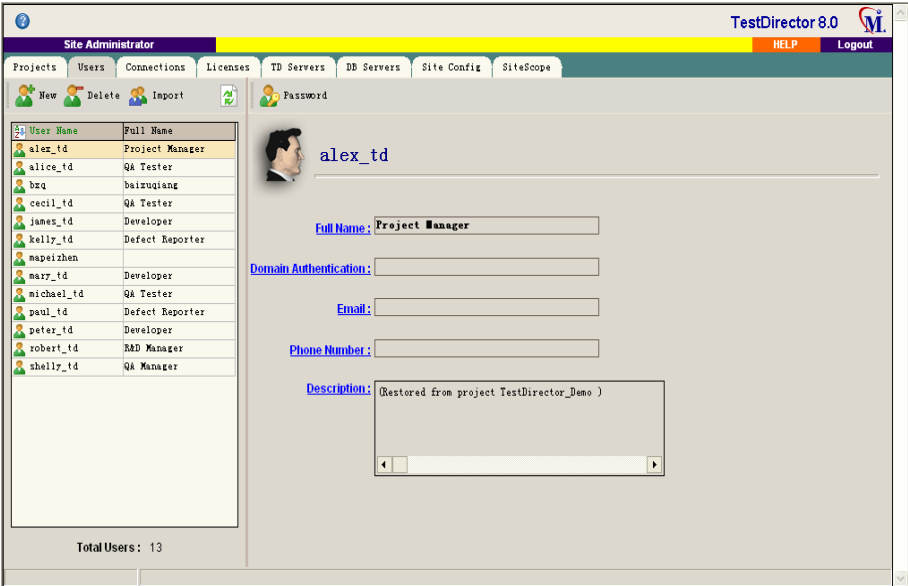
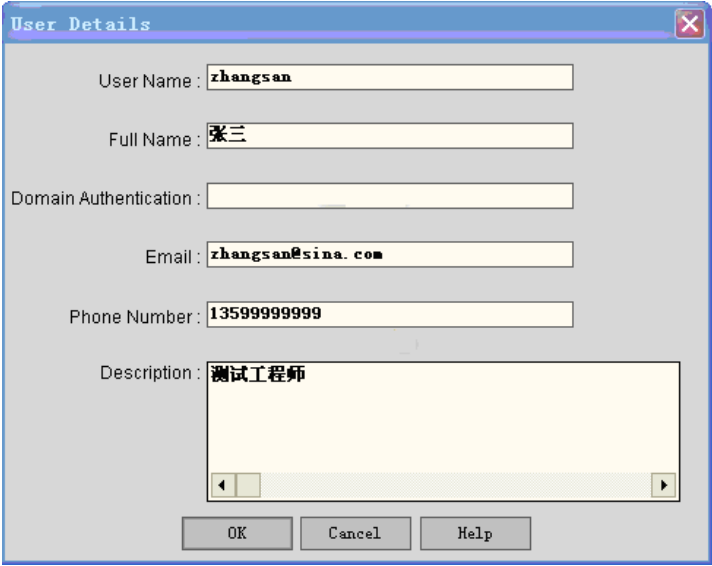


图 3- 126 用户管理界面

用户新增

- 1、在图 3- 126 中点击New。
- 2、输入用户相关信息。





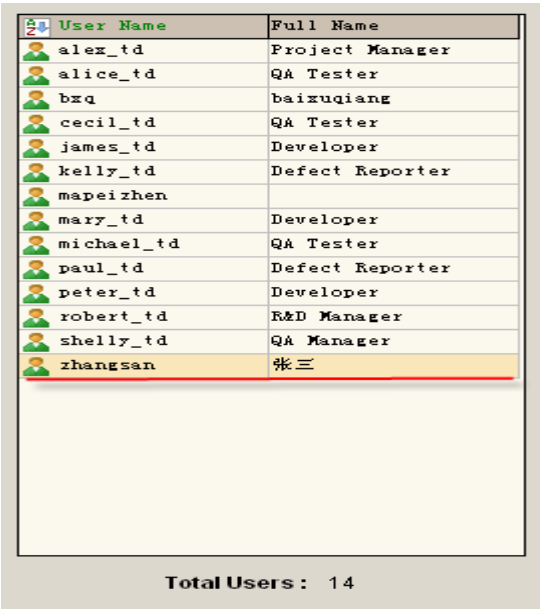
A dialog box titled "User Details" with a close button (X) in the top right corner. It contains several input fields for user information:

- User Name:
- Full Name:
- Domain Authentication:
- Email:
- Phone Number:
- Description:

At the bottom, there are three buttons: "OK", "Cancel", and "Help".

图 3- 127 用户信息添加界面

- 3、填写完毕后，点击【OK】完成用户新增。
- 4、成功界面，如 图 3- 128 所示。



A window displaying a list of users. The list has two columns: "User Name" and "Full Name". The "zhangsan" user is highlighted with a red border. Below the list, it says "Total Users : 14".

User Name	Full Name
alex_td	Project Manager
alice_td	QA Tester
bxq	baixuqiang
cecil_td	QA Tester
james_td	Developer
kelly_td	Defect Reporter
mapei zhen	
mary_td	Developer
michael_td	QA Tester
paul_td	Defect Reporter
peter_td	Developer
robert_td	R&D Manager
shelly_td	QA Manager
zhangsan	张三

Total Users : 14

图 3- 128 用户列表

- 其他用户可用同样方法新增。
- 5、设置用户登录密码。
- 点击Password，出现 图 3- 129。



图 3- 129 用户密码设置界面

输入 New Password 与 Retype Password 后，点击【OK】，完成密码设置，注意此处的密码需设置一直。

6、密码设置成功界面，如 图 3- 130 所示。



图 3- 130 密码设置成功信息提示

用户修改

1、选择需要修改的用户，如此处的 zhangsan。

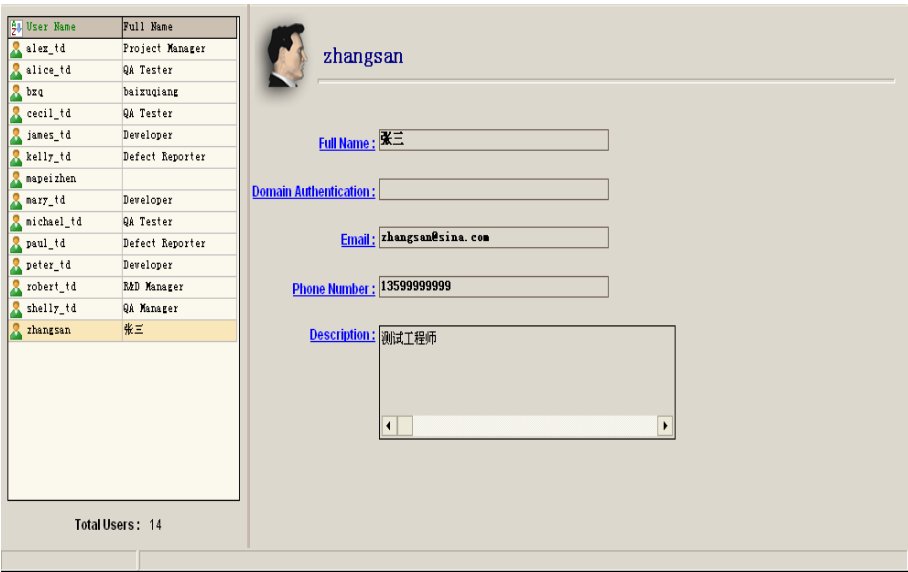


图 3- 131 用户信息

选中后，在界面的右边会出现该用户的详细信息。

2、点击“Full Name”，修改相关信息，输入需要修改的相关字段，完成后点击【OK】。

3、密码修改同新增用户操作一样。

用户删除

1、选中需删除的用户，如此处的 zhangsan。



User Name	Full Name
alex_td	Project Manager
alice_td	QA Tester
bxq	baixuqiang
cecil_td	QA Tester
james_td	Developer
kelly_td	Defect Reporter
mapeizhen	
mary_td	Developer
michael_td	QA Tester
paul_td	Defect Reporter
peter_td	Developer
robert_td	R&D Manager
shelly_td	QA Manager
zhangsan	张三

Total Users : 14

图 3- 132 选择待删除用户

2、点击Delete，出现 图 3- 133。

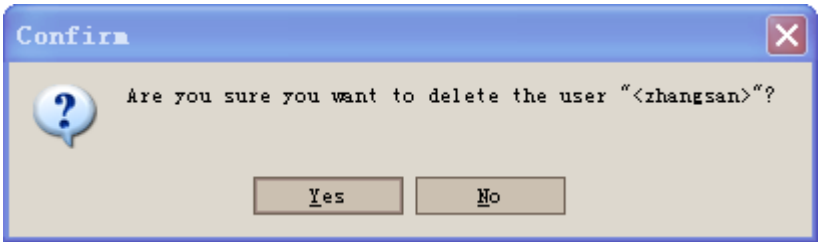


图 3- 133 删除用户确认信息

询问是否确定删除该用户，点击【Yes】，完成删除操作。

TestDirector 后台功能我们重点就学习以上几点，其它的功能并不常用，所以这里就不介绍了，学者可自行学习研究。

### 3.4.4 TestDirector 项目定制

在后台创建项目后，需要根据项目的实际情况进行相关属性的设置，比如确定组织结构、设置项目成员、制定工作流程等等。这里我们用一个例子来说明项目属性设置过程。

现假想有一项目为《新科海在线考试系统》，需要测试部门进行功能与性能的测试。分析这个测试任务，我们需要从以下几个方面来考虑。

#### 项目组织结构

首先弄清楚该项目的组织结构，由开发组，测试组构成，角色有项目经理，开发组长，开发工程师，测试组长，测试工程师。

#### 参与该项目的人员

共有多少人参与该项目，并有多少人需使用 TestDirector 进行工作。这里我们假设有 1 名项目经理，1 名开发组长，2 名开发工程师，1 名测试组长，2 名测试工程师。

#### 权限设置

这里需重点考虑的是如何控制系统的删除权限，比如删除需求，删除缺陷等。一般来说，在测试管理中是不允许执行删除操作的，但实际工作中，可能需要删除一些不必要的测试需求，测试用例等，所以在这里我们并没有严格控制这些权限，不过对于缺陷的删除操作是严禁的，在设置权限时，需要考虑这一点，一般权限设置会根据测试部门的工作规范进行设置。

#### 系统中哪些字段是必填的，哪些可以忽略

在我们所提交的缺陷中，是否有将缺陷的严重度作为必填项，发现版本作为非必填项，或者其他的要求。这个要求一般根据公司测试部门的工作规范来实现，都有事先的约定，可参照测试部门的工作规范。

在上面这些信息弄清楚后，才可以进行测试项目相关属性的设置。那么当前这个假想的项目我们抽取的信息如 表 3-4：

成员小组名称	角色名称	小组人员个数
管理组	项目经理	1
开发组	开发组长	1
	开发工程师	2
测试组	测试组长	1
	测试工程师	2

表 3-4 组织与人员结构表

### 登录项目定制

在确定被测系统相关的定制要求后，TestDirector 管理员登录到 TestDirector 后台，创建一个《新科海在线考试系统》项目，然后在项目定制功能界面进行相关信息的定制。

1、访问 [http://192.168.0.102/tdbin/start\\_a.htm](http://192.168.0.102/tdbin/start_a.htm)，进入 TestDirector 前台功能界面，如图 3-134 所示。

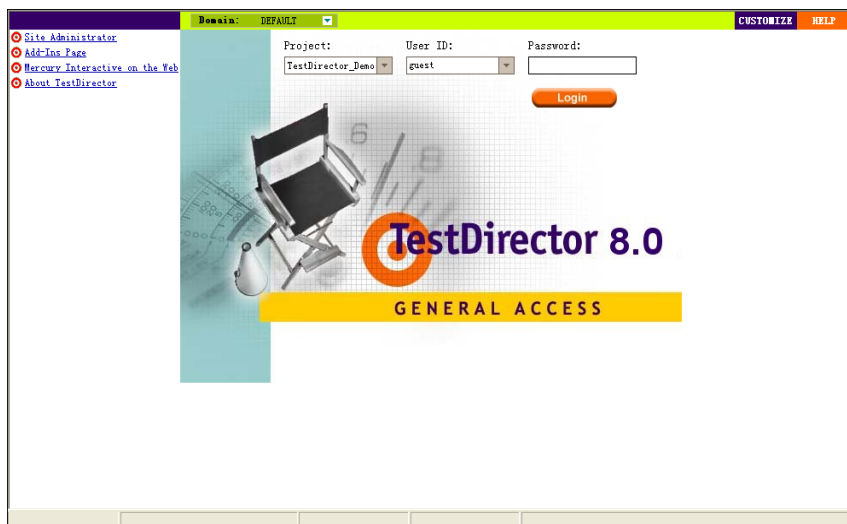


图 3-134 TestDirector 前台登录界面

2、点击 **CUSTOMIZE** 进入项目定制登录界面。



图 3-135 项目定制管理登录界面

Domain 默认, Project 选择《新科海在线考试系统》, User ID 默认为 admin, 密码为空, 完成后点击【OK】。

小提示:

TestDirector 初始管理员用户密码为空。

3、成功登录后的项目定制界面。

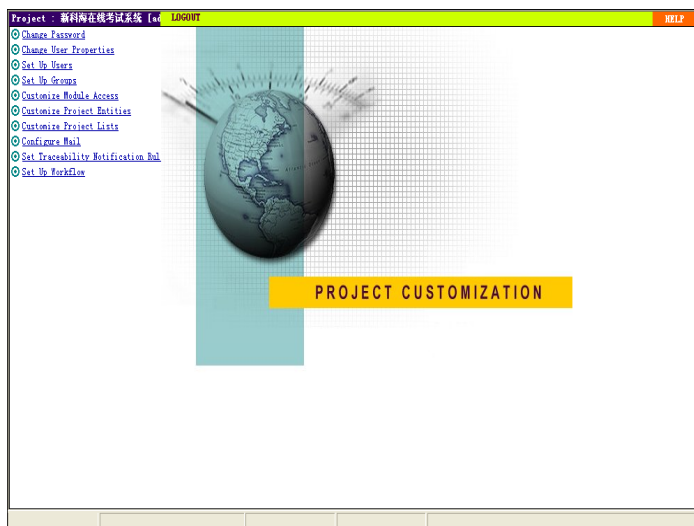


图 3-136 项目定制功能界面

修改用户密码 (Change Password)

点击“Change Password”，出现图 3-137。



图 3-137 修改用户密码界面

输入旧密码、新密码、确认密码后, 点击【OK】, 完成密码修改。

### 修改用户属性（Change User Properties）

点击Change User Properties，出现 图 3- 138。



图 3- 138 修改用户属性界面

输入相关信息后，点击【OK】，完成用户属性修改。

### 用户管理（Set Up Users）

点击Set Up Users，出现 图 3- 139。

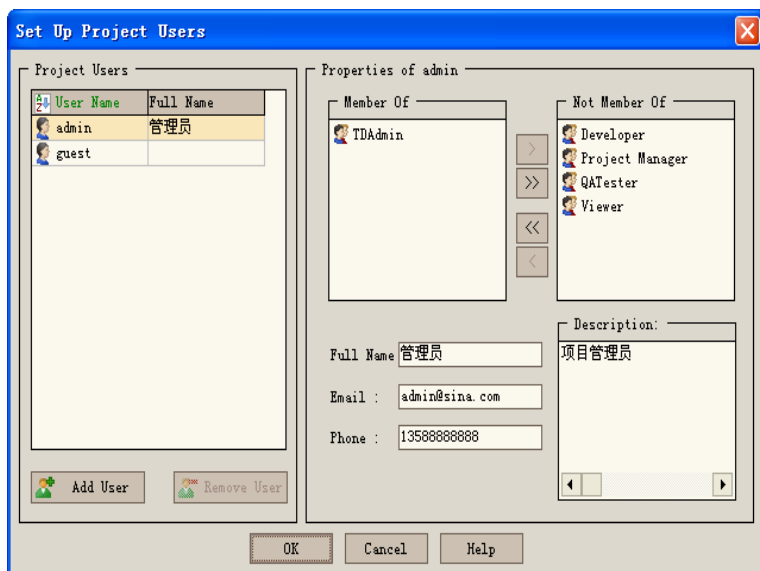


图 3- 139 用户管理界面

### 增加用户


项目默认有两个用户admin与guest，我们在实际项目管理中当然不止这两个用户，所以需要创建其他的用户，点击  Add User，出现 图 3- 140。



图 3- 140 用户列表

此处列出的用户整个 TestDirector 的用户,这些用户是在后台用户管理里维护,我们这里只能使用这些用户,如果这些用户中没有我们所需的,则可新增。

存在所需用户: 如果存在,则可选中该用户,点击【OK】后完成项目用户创建;

不存在所需用户: 点击【New】,如 图 3- 141 所示。

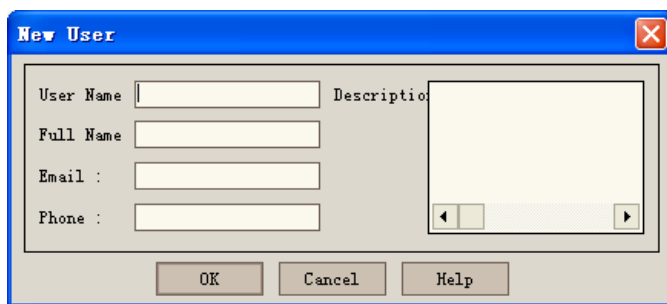


图 3- 141 添加用户功能界面

输入相关用户信息,点击【OK】完成项目用户创建,同时该用户成为 TestDirector 系统级用户。

### 用户属性维护

用户创建完成后,显示 图 3- 142。



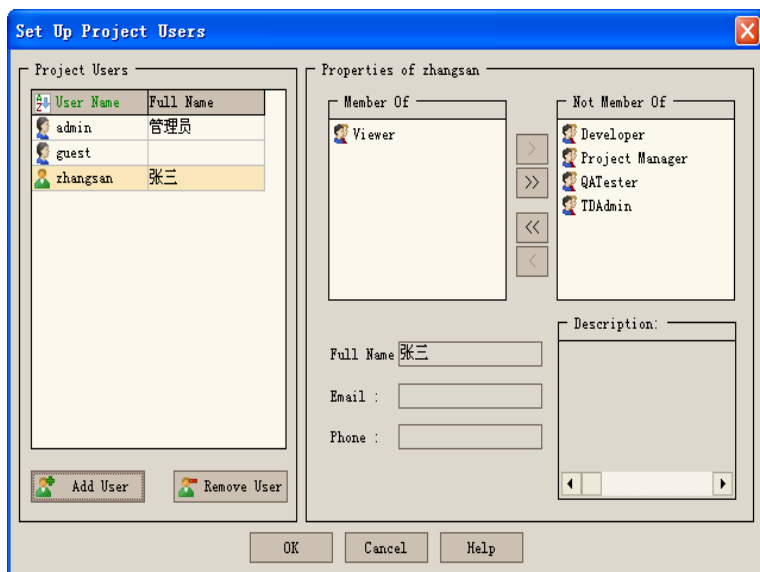




图 3-142 用户属性维护界面


该界面左边是当前项目的用户列表，右边是选中用户的属性，比如图 3-142 中的 zhangsan，它的属性是 Viewer。如果想为该用户赋予更多的角色，则可以在“Not Member Of”选择相应的角色进行创建。比如将 QATester 角色赋予 zhangsan，则选择“Not Member Of”中的“QATester”，点击 ，即可为 zhangsan 赋予 QATester 角色。


TestDirector 中定义了五种角色，Viewer、Developer、Project Manager、QATester、TDAdmin，这些角色所对应的权限是不一样的，这里简单描述：


- **Viewer:** 该组成员在 TestDirector 项目中只有只读权限，只能更改他们自身的密码和属性。
- **Developer:** 该组成员有受限制的权限在如下模块修改附件：需求、测试计划、测试库。在缺陷模块，该组成员只能创建和修改缺陷，不能删除缺陷。该组也有部分管理权力。
- **Project Manager:** 该组成员具有管理 TestDirector 项目中的需求、测试计划、测试库、缺陷模块功能，同时也具有部分系统管理功能。
- **QATester:** 该组成员具有管理 TestDirector 项目中的需求、测试计划、测试库模块功能，在缺陷模块，该组成员只能创建和修改缺陷，不能删除缺陷，该组也有部分管理权力。
- **TDAdmin:** 该组成员拥有所有权限。

注：图 3-142 中小按钮功能说明如下：

：将右边选中的项创建至左边；

: 将右边所有项创建至左边;

: 将左边选中的项创建至右边;

: 将左边所有项创建至右边

### 修改用户信息

修改用户信息的功能比较简单,选中需修改的用户,在图 3-143 中进行修改相关信息,完成后点击【OK】,即可完成修改操作。

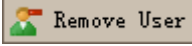


The form is titled 'Description:'. It contains three input fields on the left: 'Full Name' with the value '管理员', 'Email' with the value 'admin@sina.com', and 'Phone' with the value '13588888888'. On the right, there is a text area containing the value '项目管理员'. At the bottom right of the text area, there are two small navigation buttons, one with a left arrow and one with a right arrow.

图 3-143 修改用户信息

### 移除用户

当某个用户不工作于当前项目时,需要将该用户从当前项目中移除出去,注意,这里仅是移除,并非删除,在项目定制功能中并没有提供彻底删除用户的功能。

选中需移除的用户,如zhangsan,点击 , 出现图 3-144 所示提示。

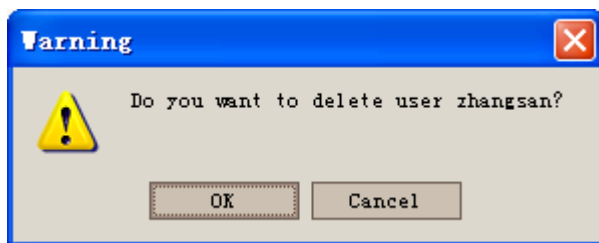


图 3-144 删除用户确认信息界面

询问是否删除用户 zhangsan? 点击【OK】,完成移除操作。移除后,该用户将不能登录该项目。

### 组管理 (Set Up Groups)

点击 Set Up Groups, 出现图 3-145。